



Grant Agreement No.: 761488



D1.4: Technical requirements (platform and service requirements)

Work package	WP 1
Task	T1.4
Due date	31/08/2018
Submission date	
Deliverable lead	ENG
Version	1.0
Authors	Ferdinando Bosco (ENG), Vincenzo Croce (ENG)
Reviewers	Stamatis Rapanakis (ATC), Vasilis Papanikolaou (ATC)
Keywords	CPN – Open Virtual Platform – Technical requirements – Technical work plan – Platform requirements - System requirements

Document Revision History

Version	Date	Description of change	List of contributor(s)
V0.1	13/07/2018	1 st version of the deliverable with table of contents	Ferdinando Bosco (ENG) Vincenzo Croce (ENG)
V0.1.1	18/07/2018	Updated table of contents and request for contributions	Ferdinando Bosco (ENG) Vincenzo Croce (ENG)
V0.2	30/07/2018	2 nd version of the deliverable with first contributions from ENG and ATC	Ferdinando Bosco (ENG) Vincenzo Croce (ENG) Marina Klitsi (ATC) Stamatis Rapanakis (ATC) Nikos Sarris (ATC)
V0.3	09/08/2018	3 rd version of the deliverable with Platform requirements	Ferdinando Bosco (ENG) Vincenzo Croce (ENG)
V0.4	21/08/2018	Draft version with contributions related to the user scenarios and pilot planning	Ferdinando Bosco (ENG) Vincenzo Croce (ENG) Tilman Wagner (DW) Olga Kisselmann (DW)
V0.4.1	29/08/2018	Reviewed version	Stamatis Rapanakis (ATC) Vasilis Papanikolaou (ATC)
V1.0	30/08/2018	Final version	Ferdinando Bosco (ENG) Vincenzo Croce (ENG)



DISCLAIMER

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 761488.

This document reflects only the authors' views and the Commission is not responsible for any use that may be made of the information it contains.

Project co-funded by the European Commission in the H2020 Programme		
Nature of the deliverable:		R
Dissemination Level		
PU	Public, fully open, e.g. web	X
CL	Classified, information as referred to in Commission Decision 2001/844/EC	
CO	Confidential to CPN project and Commission Services	



EXECUTIVE SUMMARY

This deliverable aims to gather the output from other WP1 deliverables in terms of vision, user requirements, scenarios and use cases and integrate them with technical requirements in order to have a complete technical work plan able to plan all necessary activities to meet all the objectives pursued by CPN project.

The first chapter highlights the general goals of the project, the purpose of this deliverable and in particular explains how this document not only address the standard technical requirements definition but it is also a collector of all other requirements deliverables.

The second chapter is a summary of all the other requirements collected on the other deliverables, together with user scenarios and detailed use cases. This information will be exploited to verify which are the technologies already available and in which way we need to proceed in order to completely fulfil the goals of the project.

The third chapter describes, in a detailed way, how the functionalities offered by the technology bricks already defined in D3.1 Initial Design and APIs of Technology Bricks¹, can satisfy all the user requirements gathered in WP1. This chapter define the technical work plan that address all the partners on the road to follow to improve their components and fulfil the goals of the project. In this mapping, each user requirement was mapped with the necessary technology bricks and functionalities along with the input/output expected. Particular attention has been given to the output of D1.3 that introduce some innovative components trying to understand as these can be included in the CPN platform in order to add a significant level of innovation.

The fourth chapter extends the requirements gathered until now with the platform requirements. In particular, starting from the D2.1 Reference Architecture², in which some architecture requirements was introduced, the CPN platform implementation together with its core components was described. An analysis of the components selected for the CPN architecture allowed to identify the platform requirements both functional and non-functional, going to complete the requirements collection.

Finally, the approach and the methodology used for the technical work plan was described in detail.

¹ https://www.projectcpn.eu/s/CPN_D13_Innovative-CPN-Components_20180228_V10-mngx.pdf

² <https://www.projectcpn.eu/s/D21-CPN-Reference-Architecture-v10-jydy.pdf>



TABLE OF CONTENTS

DISCLAIMER.....	3
EXECUTIVE SUMMARY	4
TABLE OF CONTENTS	5
LIST OF FIGURES.....	6
LIST OF TABLES	7
ABBREVIATIONS.....	8
1 INTRODUCTION.....	9
2 USER REQUIREMENTS	10
2.1 Summary of user requirements.....	10
2.2 User Scenarios	18
3 TECHNICAL WORK PLAN	21
3.1 Pilot 1	23
3.1.1 Description and goals	23
3.1.2 Mapping	23
3.2 Pilot 2	26
3.2.1 Description and goals	26
3.2.2 Mapping	26
3.3 Pilot 3.....	39
3.3.1 Description and goals	39
3.3.2 Mapping	39
3.4 Innovative components Analysis	46
3.4.1 User Scenarios	48
3.4.2 Technology Bricks	49
4 PLATFORM REQUIREMENTS	51
4.1 CPN Platform	51
4.1.1 Container management platform.....	51
4.1.2 Core components	53
4.1.3 CPN Catalog and Registry	57
4.2 Functional requirements	59
4.3 Non-Functional requirements.....	59
5 APPROACH AND METHODOLOGY	62
6 CONCLUSIONS	65



LIST OF FIGURES

FIGURE 1: CPN PLATFORM ARCHITECTURE51

FIGURE 2: CPN API GATEWAY GUI.....55

FIGURE 3: TWO WAYS FOR IMPLEMENTING ORCHESTRATION56

FIGURE 4: CPN CATALOG EXAMPLE.....58

FIGURE 5: CPN IMPLEMENTATION APPROACH.....62

FIGURE 6: LONGITUDINAL RESEARCH DESIGN WITH INTER-CASE BENCHMARK.....62

FIGURE 7: CPN TRELLO BOARD.....64

LIST OF TABLES

TABLE 1: USER PROFILE REQUIREMENTS	14
TABLE 2: APPLICATION FEATURES REQUIREMENTS	17
TABLE 3: PRODUCTION SIDE REQUIREMENTS	18
TABLE 4: THE DEFINITIVE LIST OF CPN TECHNOLOGY BRICKS.....	23
TABLE 5: PILOT 1 – MAPPING BETWEEN USER REQUIREMENTS AND FUNCTIONALITIES	26
TABLE 6: PILOT 2 – MAPPING BETWEEN USER REQUIREMENTS AND FUNCTIONALITIES	39
TABLE 7: PILOT 3 – MAPPING BETWEEN USER REQUIREMENTS AND FUNCTIONALITIES	46
TABLE 8: CPN PLATFORM FUNCTIONAL REQUIREMENTS	59
TABLE 9: CPN PLATFORM NON-FUNCTIONAL REQUIREMENTS	61



ABBREVIATIONS

ATC	Athens Technology Center
CPN	Content Personalisation Network
DCAT	Digital Catapult
e.g.	Example given
ENG	Engineering Ingegneria Informatica
Etc.	Etcetera
GDPR	General Data Protection Regulation
GUI	Graphical User Interface
Imec	Interuniversity MicroElectronics Center
JSON	JavaScript Object Notation
JWT	JSON Web Token
QoS	Quality of Service
RDF	Resource Description Framework
RSS	RDF Site summary
UI	User interface
UR	User requirement
YAML	YAML Ain't Markup Language



1 INTRODUCTION

A technical requirement pertains to the technical aspects that your system must fulfil, such as performance-related issues, reliability issues, and availability issues. These types of requirements are often called quality of service (QoS) requirements, service-level requirements or non-functional requirements.³

This document not only describes the standard technical requirements necessary to implement the CPN solution but also studies the output produced by the other WP1 tasks.

In fact, it aims to provide a complete overview of all the requirements collected, in order to have a single document that defines the road to follow to achieve the goals that the CPN project has set itself. To do this it was necessary to define a detailed technical work plan, in agreement among all the partners that allows evolving the technologies initially provided by the technological bricks of CPN for advancing the current state of the art in term of content personalisation.

User requirements, scenarios and use cases are necessary to understand how technology bricks fit into the CPN solution, how they can meet some of the requirements and how they can evolve to meet others.

The system requirements, as well as the standard technical (non-functional) requirements related to scalability, performance and system availability level, also describe the requirements for the creation of CPN the open virtual platform, which is another fundamental element of the project's innovation. In fact, this platform allows not only setting up a solution that meets the requirements collected, but also to be extended and configured to introduce new components and new features, as described in the D2.1 CPN Reference Architecture.

The user requirements, together with the requirements of the open virtual platform CPN, both functional and non-functional, are therefore to be placed as the basis for the start of development activities, as defined in the technical work plan, and to achieve the level of innovation necessary to make the CPN solution a fundamental tool for both end-users and professional users.

³ <http://agilemodeling.com/artifacts/technicalRequirement.htm>

2 USER REQUIREMENTS

The deliverable D1.1 User Requirements⁴ shows the detailed process of the requirements collection of the CPN project.

It starts with an analysis of the already existing personalisation systems examining the stage of personalisation in each media organisation involved into the project as well as the possible challenges and opportunities for public and private broadcasters.

A particular point of attention was dedicated to the GDPR guidelines, incorporating them by design from the beginning.

Furthermore, many extensive evaluation activities were conducted, among which interviews with media partners, co-creation sessions and online surveys.

As output of these activities, some personas were identified and a list of user requirements, divided into three categories:

- ➔ **User Profile**, collects all data related to the user and his/her interests, mood or other attributes that build the basis for the personalisation process on the user's end
- ➔ **Application features**, new additional features to make a new and unique way for news consumption
- ➔ **Production Side**, features focussed on the producer of contents

In the paragraphs below, we report a summary of the user requirements collected, which will be the basis for the elaboration of the technical work plan and the mapping with technology bricks functionalities.

2.1 SUMMARY OF USER REQUIREMENTS

User Profile	
ID	Requirements

⁴ https://www.projectcpn.eu/s/CPN_D11_User-Requirements_201802028_V10.pdf



UR-UP1 - Interests (Categories, Entities, and Values): What topics is the user interested in?	
UR-UP1.1	The system must allow the users to manually choose their interests that later define the personalisation
UR-UP1.2	The system should create/refine interests based on the user's consumption habits
UR-UP1.3	The system should be able to offer personalised content on the basis of the users mood or values
UR-UP1.4	The system should refine the user's interests through frequent interaction with the user (talkback)
UR-UP1.5	The system should refine the interests based on the user's behaviour on social networks (through data upload or connection of the networks)
UR-UP1.6	The system should assign preferences (1-5) to categories based on the users behaviour
UR-UP1.7	The system should allow users to assign and change preferences (1-5) to categories themselves
UR-UP1.8	The system must allow users to completely turn off the personalisation algorithm and receive content as is and vice versa
UR-UP2 - Network: Making use of connections the user already has through social media	
UR-UP2.1	The system should allow for social media integration to recommend content based on what connections like, read and share
UR-UP2.2	The system should offer a recommendation of articles based on most liked/most shared numbers from a user's network and beyond that. (Nuzzle-Feature)
UR-UP2.3	The system should allow for social media integration to keep track of what the user has already seen elsewhere.
UR-UP2.4	The system should be able to analyse whom a user has been most interacting with on social media to prioritize the users for the personalisation
UR-UP2.5	The system should allow the user to down-/upload their network connections through user account.
UR-UP2.6	The system should allow users to search for other users on social media to build direct connections



UR-UP2.7	The system should allow users to share content from the CPN system to social networks
UR-UP3 - Time & Length: When does the user prefer to consume content and for how long?	
UR-UP3.1	The system must allow the user to choose a preferred time frame or frames to consume content
UR-UP3.2	The system should create/refine time frames based on the user's consumption habits
UR-UP3.3	The system should refine the user's time frames through frequent interaction with the user (talkback)
UR-UP3.4	The system should use the time frames in order to decide how many items of what length and of what format it offers to the user
UR-UP3.5	The system must allow the user to postpone a time frame for a chosen amount of time.
UR-UP3.6	The system must allow the user to ignore a time frame completely
UR-UP3.7	The system should learn from these user responses and adjust its offerings accordingly
UR-UP4 - Preferred Media: Which type of content does the user prefer?	
UR-UP4.1	The system must allow the user to choose preferred types of content
UR-UP4.2	The system should set/refine preferred types of content based on the user's consumption habits and the timing
UR-UP4.3	The system should refine the user's preferred types of content through frequent interaction with the user (talkback)
UR-UP5 - Location & Surroundings: Where is the user and what's going on around him/her?	
UR-UP5.1	The system should make use of the location data of the user (permission of the user granted) to choose the right content for the user
UR-UP5.2	The system should allow the user to set a home/main interest location
UR-UP5.3	The system should make use of the location data of the user to determine the best point in time to offer content



UR-UP5.4	The system should try to determine the surroundings of the user based on either just location data or location data and direct interaction with the user (talkback)
UR-UP5.5	The system must give the user an easy option to agree to or withdraw from using location data for personalised offers
UR-UP6 - Knowledge (Management): What does the user already know?	
UR-UP6.1	The system must keep track of what content the user has already consumed on a piece and on a content basis within CPN and beyond
UR-UP6.2	The system must keep track of how much of each item users consume, where they stop, continue and what they skip
UR-UP6.3	The system should interact with the user in order to refine user interests in regards to why something was skipped or something was consumed completely
UR-UP6.4	The system should be able to offer insights and advice based on what it learned about what a user consumed in relation to a certain entity (e.g. a place)
UR-UP6.5	The system should allow the user to delete part of the systems knowledge for specific time frames back in time from the moment of viewing
UR-UP7 - Devices: On what device is the user consuming content?	
UR-UP7.1	The system should check on what device the user is consuming the content
UR-UP7.2	The system should adjust its content offering based on the type of device the user is using
UR-UP7.3	The system should try to make smart use of device data to determine the surroundings of the user and adjust the content strategy accordingly
UR-UP8 - Importance for user: What is relevant for the user, outside their given interests?	
UR-UP8.1	The system should combine reading habits and knowledge about the user to provide smart updates on things the user could be interested in, even if this doesn't fit his/her set interests
UR-UP8.2	The system should always offer content that has a direct influence on the users (e.g. life-threatening), overruling other interest settings
UR-UP8.3	The system should be able to surprise the user with content, he/she would not have chosen themselves



UR-UP9 - User Profile Management: Giving the user transparency and control over their Data	
UR-UP9.1	The system must provide transparent, simple and easy-to-understand information on what user data are collected, for what purpose and how they are stored
UR-UP9.2	The system should require informed and explicit consent for processing of personal user data, beyond those required for the provisioning of the agreed service
UR-UP9.3	The system must give the user a full overview of his/her data and allow them full control, including update and removal of data
UR-UP9.4	The user must be able to change and overwrite settings in their profile
UR-UP9.5	The user must be able to download their profile data in CPN in a machine readable format and a user friendly format
UR-UP9.6	The system should allow the user to add external data to update their profile

Table 1: User Profile requirements

Application features	
ID	Requirements



UR-AF1 - Bursting the Filter Bubble: How can CPN avoid filter bubbles and echo chambers?	
UR-AF1.1	The system should offer users an overview of other sources, covering the same topic
UR-AF1.2	The system should highlight differences between the perspectives of different sources on a similar topic
UR-AF1.3	The system should offer the user an easy overview of what content from which sources he has consumed over a certain period of time
UR-AF1.4	The system should make it easy for the user to see a bias of a content item or source
UR-AF1.5	The system should allow users to choose favorite sources
UR-AF1.6	The system should offer the user a random news selection upon request based on certain data and preferences of the users profile, which the user can choose
UR-AF2 - Avoiding FOMO: How to ensure people think they know everything there is to know?	
UR-AF2.1	The system should show users who else from their network has consumed the same content item.
UR-AF2.2	The system should show users what else their network has shown, if there are differences
UR-AF2.3	The system should be able to show users the content item from another user (anonymously)
UR-AF2.4	The system should show users only a limited number of items at once
UR-AF2.5	Once all articles proposed have been consumed, the system should only offer more content upon request by the users
UR-AF3 - Content/Format: In which way do we have to prepare content for the user?	
UR-AF3.1	The system should offer content items in small, easy to consume and logical packages, allowing the user to consume them bit by bit
UR-AF3.2	The system should offer the user a short overview of all important headlines at a specific point in time with access to more details upon request



UR-AF3.3	The system should allow users to choose whether they prefer an overview or all content at once
UR-AF3.4	The system should be able to offer both news content and entertainment
UR-AF3.5	The system should be able to offer both locally and globally relevant content
UR-AF3.6	The system should be able to put global news in a local relevance context for users
UR-AF3.7	The system should be able to give the user a timeline overview of events regarding a specific topic
UR-AF3.8	The system should allow users to filter content by language
UR-AF3.9	The system should allow users to filter content by complexity within a language
UR-AF4 - Sources: Where does the necessary content come from?	
UR-AF4.1	The system should be able to personalise news from/for the CPN media partners (VRT, DIAS, DW)
UR-AF4.2	The system should allow for additional content sources, outside the consortium
UR-AF5 - Transparency: Giving the user control & understanding over the content he sees.	
UR-AF5.1	The system must offer the user an easy to access and easy to understand overview of their profile
UR-AF5.2	The system must offer users easy access to their profile in order to change settings and data
UR-AF5.3	The system must make it transparent to the users why they are shown certain content, based on an item level
UR-AF6 - Archive: Making content available beyond the moment	
UR-AF6.1	The system must allow users to access content again that they have already opened before
UR-AF6.2	The system should allow users to consume content beyond their predefined timeframe after an interaction with the user (talkback)
UR-AF6.3	The system should allow users to actively save articles for later consumption



UR-AF6.4	The system should be able to memorize where a user left off and restart at the same point
UR-AF7 - User Feedback: Asking users to help improve the system	
UR-AF7.1	The system should offer user feedback requests in a playful/entertaining way
UR-AF7.2	The system should include guided feedback for specific elements of the system, allowing users to (help) improve it
UR-AF7.3	The system should allow users to assign both existing or new attributes (categories, moods etc.) to a content item
UR-AF7.4	The system should be able to offer a feedback interaction to determine the ground level of personalisation based on mood, time and interest
UR-AF8 - Temporary Categories: Users can temporarily change the personalisation algorithm	
UR-AF8.1	The system should allow users to search for specific topics they are temporarily interested in
UR-AF8.2	The system should allow users to add this search as a temporary personalisation category
UR-AF8.3	The system should allow users to define a specific time frame for this temporary change
UR-AF9 - Mute topics: Exclude topics from the personalisation for a certain time	
UR-AF9.1	The system should allow users to define keywords and logical combinations of them to exclude content from their personalisation
UR-AF9.2	The system should allow users to define a time frame per keyword/logical combination
UR-AF9.3	The system should be able to overwrite this exclusion for important breaking rules

Table 2: Application features requirements

Production side	
ID	Requirements



UR-PS1 - Detailed Analytics: Giving newsrooms a more detailed feedback on their audience	
UR-PS1.1	The system should show the access to items through users by numbers (who, when, how long)
UR-PS1.2	The system should show which parts (paragraphs, entities) of an item were most interesting to users
UR-PS1.3	The system should show which topics were most interesting to users
UR-PS1.4	The system should be able to show these numbers during the creation process of the content
UR-PS2 - Integration: How should CPN be connected to the production side?	
UR-PS2.1	The system should allow for an easy integration into the producers workflow
UR-PS2.2	The system should provide contract templates to allow freelancers to easily work together and with editors, to define and track the scope of individual contributions and expected revenues
UR-PS2.3	The system should allow producers to transparently see how often their contributions are used and distributed to readers
UR-PS2.4	The system should allow producers to export the record of their publications through standardized and interoperable formats
UR-PS2.5	The system should allow for an easy contribution of content from different publishers through standardised interfaces
UR-PS2.6	The system should give feedback on what attributes are best used on content to improve the personalisation performance
UR-PS2.7	The system should allow editors to easily add missing attributes to articles manually

Table 3: Production side requirements

2.2 USER SCENARIOS

The deliverable D1.2 User Scenarios describes the updated version of the User Scenarios, used as blueprints to guide the development and the detailed description of the process from which the project evolved from basic user requirements to an actual application design.



In this paragraph we report a complete list of the User Scenarios (with a brief description of each of them) in order to have a complete overview of the requirements.

User Scenario 1: The Power News User

The first User Scenario is based on the initial User Scenario of a frequent and critical news consumer as described in the DoW. The initial assumptions of usage patterns and challenges of news consumption have been validated with the help of previous work as described in D1.1. These assumptions provide the basis of this initial user Scenario and should be addressed through the project. Here we intend to address critical, high-volume news consumers, who are interested in a broad array of news content but feel a sense of frustration with the current situation. This Use-Case shall carefully address various aspects of online news consumption, which may have a negative impact on the overall news experience. These aspects include information overload, fear of missing out, as well as lack of control, transparency and real and perceived filter bubbles.

User Scenario 2: The Data-Minded Sceptic

The Second User Scenario is also derived from the DoW and especially addresses the needs of a User-Group that might be overly sceptical about news personalisation and in particular of the usage of their data and watches closely in which way companies address the customers concerns and implement GDPR policies. To cater for this group transparency, control and data protection settings must be especially highlighted and therefore create a more trustful environment for this particular user group that might otherwise shun personalised news applications. At the other hand, this Use-Case also addresses the primal requirements of the platform, as of being compatible with the GDPR.

User Scenario 3: The Light News User

The Third User-Scenario represents the needs of an infrequent news consumer and is a further development of the Light-News User described in D1.1. This User-Scenario caters to a user group with reduced news consumption routines which may prioritise content from a single trustful source like a public broadcaster or their favourite local or international private news organisation. The infrequent user still wants essential updates and may value innovative features that adjust content to their current situation and may use different data sources for wholesome personalisation.

Addressing their needs is an essential task for media providers who intend to enlarge their audience, revenue and support the civic assignment of in case of public broadcasters.

User-Scenario 4: The Product Manager's View

The fourth User-Scenario enters into the equation the perspective of the production site, the editors and product manager responsible for the deployments of new technologies at media enterprises. As stated in the DoW, the CPN platform must provide value to large and small news organisations alike and simplify the process of content personalisation.

User Scenario 5: The freelancer's view



The fifth User Scenario will address individual editors and content creators, and add innovative and useful aspects to content licensing. This scenario focuses on the idea of incorporating a distributed ledger technology for licensing media content. This would allow freelancers to better track their work and help them to get adequate payment in return. The details of this are described in full in the deliverable D1.5 Content Licensing and Distribution framework, which also contains a long version of the use cases, foreseen for this scenario.



3 TECHNICAL WORK PLAN

Once the requirements were defined, it was necessary to define a technical work plan, shared by all the partners, that would allow the implementation of the CPN platform.

CPN will create a user-driven solution that permits to refine and evolve the news personalisation. In order to have a continuing increment of functionalities, it was important to implement an iterative approach and in particular with agile SCRUM methodologies⁵.

The iterative approach consists of three phases. At the end of each phase, a prototype is produced. Based on these prototypes, a series of pilots will be conducted, which would allow to collect users' feedback through tools to test, validate, and increase the functionality offered by the CPN solution at each subsequent stage. Further details about the approach and the methodologies used will be given in chapter 5.

For each pilot a very precise scope has been defined and a series of objectives to be reached, in order to be able to divide the user requirements among the various drivers in a homogeneous and functional way.

The task of dividing the requirements and identifying the modules necessary to satisfy these requirements was carried out within the deliverable D3.1. This activity has been extended adding a detailed mapping of the functionalities of each single technology bricks with the user requirements and an expected input / output list.

Below is reported the definitive list of technology bricks that will be released within the CPN project:

Layer	Technology Bricks	Planned	Partner responsible of the provision
Content	Semantic Lifting	Pilot 2	Imec
	Relation Extraction	Pilot 1	Imec
	Topic Extractor	Pilot 2	LiveTech
	Uplifting/Depressing Article Classifier	Pilot 3	Imec
	Frame Based Slot-Filling System	Pilot 2	Imec
	Sentiment	Pilot 2	LiveTech
User	UserModelling	Pilot 1	LiveTech
	Reader's App	Pilot 1	ATC
	Personal Data Receipts	Pilot 1	DigiCat
Mapping	Producer's App	Pilot 1	ENG
	Reward Framework	Pilot 3	DigiCat
	Twitter Analytics	Pilot 2	ATC
	Recommender	Pilot 1	LiveTech

⁵ <http://scrummethodology.com/>



Table 4: The definitive list of CPN technology bricks

3.1 PILOT 1

3.1.1 Description and goals

The first pilot aims to demonstrate a basic recommendation system that combines all the source providers of CPN solution (DW, VRT, DIAS) and integrates a first version of technology bricks.

During this pilot, a web application will be provided to gather user feedback on recommender module and technical interoperability and an app demonstrator to gather feedback on CPN design and features.

3.1.2 Mapping



UR	Technology (Partner)	Brick	Functionality	Input/ Output
UP1.2	The system should create/refine interests based on the user’s consumption habits			
	Reader’s App (ATC)		Track user actions	In -> UI Out -> User Actions
			Provide user signup and login	In -> UI Out -> User login information
			Collect user personal data information	In -> UI Out -> User personal data
	User Modelling (LiveTech)	Create and refine interests	In -> User actions Out -> User interests	
UP 1.8	The system must allow users to completely turn off the personalisation algorithm and receive content as is and vice versa			
	Reader's App (ATC)		Show also a stream of content which is not personalised	In -> Content Out -> Show all content on UI
UP 2.7	The system should allow users to share content from the CPN system to social networks			
	Reader's App (ATC)		Allow users to share content to social networks	In -> UI Out -> Content sent to SNs
UP 5.2	The system should allow the user to set a home/main interest location			
	Reader's App (ATC)		support setting a home/main interest location	In -> UI Out -> interest location
	User Modelling (LiveTech)		Update user model based on home/main interest location	In -> interest location Out -> Updated user model



UP 9.1	The system must provide transparent, simple and easy-to-understand information on what user data are collected, for what purpose and how they are stored		
	Reader's App (ATC)	Provide option for requesting user data information	In -> UI Out -> request data receipt
	Personal Data Receipts (DCAT)	provide an API to retrieve a user's record from User Modelling and display user data	In -> request of data receipt Out -> data receipt
	User Modelling (LiveTech)	Provide User Data record	In -> request of data receipt Out -> User data record
UP 9.2	The system should require informed and explicit consent for processing of personal user data, beyond those required for the provisioning of the agreed service		
	Reader's App (ATC)	Request user consent when user registers	In -> UI Out -> request user consent
	Personal Data Receipts (DCAT)	provide form requesting user consent based on the data to be kept in the user model	In -> request for user consent Out -> user consent form
	User Modelling (LiveTech)	Provide User Data record	In -> request of user data record Out -> User data record
	Personal Data Receipts (DCAT)	Update user data model based on the consent received	In -> user consent Out -> user model update
AF 2.4	The system should show users only a limited number of items at once		
	Reader's App (ATC)	Show users only a limited number of items at once	In -> Content items Out -> UI
AF 2.5	Once all articles proposed have been consumed, the system should only offer more content upon request by the users		



	Reader's App (ATC)	Offer more content once all proposed articles have been consumed	In -> Content items Out -> UI
AF 4.1	The system should be able to personalise news from/for the CPN media partners (VRT, DIAS, DW)		
	Recommender (LiveTech)	Must filter content based on user model	In -> content Out -> recommended content
AF 7.2	The system should include guided feedback for specific elements of the system, allowing users to (help) improve it		
	Reader's App (ATC)	Support sending user feedback	In -> UI Out -> user feedback

Table 5: Pilot 1 – Mapping between user requirements and functionalities

3.2 PILOT 2

3.2.1 Description and goals

The second pilot aims to demonstrate the overall CPN functionalities and evaluate the user experience during the use of the mobile application, the quality of the recommendations, the GDPR compliance and the first version of production side analytics.

During this pilot, a mobile application will be provided in order to gather user feedback on all the goals above listed.

3.2.2 Mapping



UR	Technology (Partner)	Brick	Functionality	Input/ Output
UP 1.4	The system should refine the user's interests through frequent interaction with the user (talkback)			
	Reader's App (ATC)		Support interaction to refine user's interests (talkback)	In -> UI Out -> User Actions
	User Modelling (LiveTech)		Refine user's interests	In -> User actions Out -> User interests
UP 1.5	The system should refine the interests based on the user's behaviour on social networks (through data upload or connection of the networks)			
	Reader's App (ATC)		Get user profile info from TruthNest	In -> Social media account Out -> Social profile
	User Modelling (LiveTech)		Update user model based on the user's social profile	In -> Social profile Out -> User interests
UP 1.6	The system should assign preferences (1-5) to categories based on the users behaviour			
	User Modelling (LiveTech)		Assign preferences to categories after collecting user actions	In -> User actions Out -> User preferences
UP 1.7	The system should allow users to assign and change preferences (1-5) to categories themselves			
	Reader's App (ATC)		Display category preferences and allow the user to edit them.	In -> UI Out -> User preferences
	User Modelling (LiveTech)		Update user model based on the user preferences	In -> User preferences Out -> Updated user model
UP 2.1	The system should allow for social media integration to recommend content based on what connections like, read and share			



	Reader's App (ATC)	Get user profile info from TruthNest	In -> Social media account Out -> Social profile
	User Modelling (LiveTech)	Update user model based on the user's social profile	In -> Social profile Out -> User interests
UP 2.2	The system should offer a recommendation of articles based on most liked/most shared numbers from a user's network and beyond that. (Nuzzle-Feature)		
	Reader's App (ATC)	Get user profile info from TruthNest	In -> Social media account Out -> Social profile
	User Modelling (LiveTech)	Update user model based on the user's social profile	In -> Social profile Out -> User interests
UP 2.3	The system should allow for social media integration to keep track of what the user has already seen elsewhere.		
	Reader's App (ATC)	Get user profile info from TruthNest	In -> Social media account Out -> Social profile
	User Modelling (LiveTech)	Update user model based on the user's social profile	In -> Social profile Out -> User interests
UP 2.4	The system should be able to analyse whom a user has been most interacting with on social media to prioritize the users for the personalisation on social media to prioritize the users for the personalisation		
	Reader's App (ATC)	Get user profile info from TruthNest	In -> Social media account Out -> Social profile
	User Modelling (LiveTech)	Update user model based on the user's social profile	In -> Social profile Out -> User interests
UP 2.5	The system should allow the user to down-/upload their network connections through user account		



	Reader's App (ATC)	Import/ export related data on a popular format	In -> Network connections/ file Out -> file/ network connections
UP 2.6	The system should allow users to search for other users on social media to build direct connections		
	Reader's App (ATC)	Use TruthNest to search on social media	In -> User name Out -> User
	User Modelling (LiveTech)	Update user connections	In -> User connections Out -> Updated user model
UP 3.1	The system must allow the user to choose a preferred time frame or frames to consume content		
	Reader's App (ATC)	Allow the user to choose a preferred time frame or frames to consume content.	In -> UI Out -> Time frame preferences
	User Modelling (LiveTech)	Update user model based on time frame preferences	In -> Time frame preferences Out -> Updated user model
UP 3.2	The system should create/refine time frames based on the user's consumption habits		
	Reader's App (ATC)	Track temporal user behaviour	In -> UI Out -> User temporal activity
	User Modelling (LiveTech)	Track user consumption habits	In -> User temporal activity Out -> Updated user model
UP 3.3	The system should refine the user's time frames through frequent interaction with the user (talkback)		



	Reader's App (ATC)	Allow the user to revise the preferred time frame or frames to consume content.	In -> UI Out -> Time frame preferences
	User Modelling (LiveTech)	Update user model based on time frame preferences	In -> Time frame preferences Out -> Updated user model
UP 3.4	The system should use the time frames in order to decide how many items of what length and of what format it offers to the user length and of what format it offers to the user		
	Reader's App (ATC)	Allow the user to revise the preferred time frame length and format	In -> UI Out -> Time frame preferences
	User Modelling (LiveTech)	Update user model based on time frame preferences	In -> Time frame preferences Out -> Updated user model
UP 3.5	The system must allow the user to postpone a time frame for a chosen amount of time.		
	Reader's App (ATC)	Allow the user to revise the preferred time frame or frames to consume content.	In -> UI Out -> Time frame preferences
	User Modelling (LiveTech)	Update user model based on time frame preferences	In -> Time frame preferences Out -> Updated user model
UP 3.6	The system must allow the user to ignore a time frame completely		
	Reader's App (ATC)	Allow the user to revise the preferred time frame or frames to consume content.	In -> UI Out -> Time frame preferences



	User Modelling (LiveTech)	Update user model based on time frame preferences	In -> Time frame preferences Out -> Updated user model
UP 3.7	The system should learn from these user responses and adjust its offerings accordingly		
	Reader's App (ATC)	Track user responses	In -> UI Out -> User responses
	User Modelling (LiveTech)	Update user model based on user responses	In -> User responses Out -> Updated user model
UP 5.1	The system should make use of the location data of the user (permission of the user granted) to choose the right content for the user		
	Reader's App (ATC)	Ask user for permission to use location data.	In -> UI Out -> Permission to use data
	Personal Data Receipts (DCAT)	provide an API to update the location permissions of a user	In -> request of data receipt Out -> location permissions update
	User Modelling (LiveTech)	Update user model based on location preferences	In -> Location preferences Out -> Updated user model
UP 5.3	The system should make use of the location data of the user to determine the best point in time to offer content		
	User Modelling (LiveTech)	Provide an API to get the location data of the user	In -> request for Location data Out -> Location data
	Recommender (LiveTech)	Must filter content based on location data.	In -> Location data Out -> recommended content



UP 5.4	The system should try to determine the surroundings of the user based on either just location data or location data and direct interaction with the user (talkback)		
	Reader's App (ATC)	Track location data.	In -> Device's location data/ UI Out -> Location data
	Producer's App (ENG)	Save location information of content	In -> location information Out -> CPN content
	User Modelling (LiveTech)	Update user model based on location preferences	In -> Location preferences Out -> Updated user model
UP 5.5	The system must give the user an easy option to agree to or withdraw from using location data for personalised offers		
	Reader's App (ATC)	Ask user for permission to use location data.	In -> UI Out -> Permission to use data
	Personal Data Receipts (DCAT)	provide an API to update the permissions	In -> request of data receipt Out -> permissions update
UP 6.1	The system must keep track of what content the user has already consumed on a piece and on a content basis within CPN and beyond		
	Reader's App (ATC)	Track user actions and trigger respective events.	In -> User actions explicitly tracked Out -> Save the actions
	User Modelling (LiveTech)	Update user model based on user actions	In -> User actions Out -> Updated user model
UP 6.2	The system must keep track of how much of each item users consume, where they stop, continue and what they skip		
	Reader's App (ATC)	Track user actions and trigger respective events.	In -> User actions explicitly tracked Out -> Save the actions



	User Modelling (LiveTech)	Update user model based on user actions	In -> User actions Out -> Updated user model
UP 6.3	The system should interact with the user in order to refine user interests in regards to why something was skipped or something was consumed completely		
	Reader's App (ATC)	Ask user questions	In -> UI Out -> Answers to questions
	User Modelling (LiveTech)	Update user model based on user actions	In -> User actions Out -> Updated user model
UP 8.2	The system should always offer content that has a direct influence on the users (e.g. life-threatening), overruling other interest settings		
	Reader's App (ATC)	Show notifications	In -> Read notifications (asynchronously) Out -> Show notifications
	Producer's App (ENG)	Annotate content as "breaking news"	In -> outside sources Out -> CPN content
	Recommender (LiveTech)	Implement "breaking news" functionality.	In -> "Breaking news" content Out -> recommended content
UP 9.3	The system must give the user a full overview of his/her data and allow them full control, including update and removal of data		
	Reader's App (ATC)	Show user data and edit/ delete the data	In -> UI Out -> Edited data
	User Modelling (LiveTech)	Update user model based on user data	In -> User data Out -> Updated user model
UP 9.4	The user must be able to change and overwrite settings in their profile		



	Reader's App (ATC)	Show/ edit user settings	In -> UI Out -> Edited user data
	User Modelling (LiveTech)	Update user model based on user data	In -> User data Out -> Updated user model
UP 9.5	The user must be able to download their profile data in CPN in a machine readable format and a user friendly format		
	Reader's App (ATC)	Extract json file	In -> UI Out -> Json data
AF 1.5	The system should allow users to choose favourite sources		
	Reader's App (ATC)	Allow users to choose favourite sources.	In -> UI Out -> favourite sources
	User Modelling (LiveTech)	Update user model	In -> favourite sources Out -> user model update
AF 1.6	The system should offer the user a random news selection upon request based on certain data and preferences of the users profile, which the user can choose		
	Reader's App (ATC)	Show a stream of randomly created news. Design forms with filters.	In -> UI Out -> news stream
	Recommender (LiveTech)	Offer content based on criteria	In -> Criteria Out -> News selection
AF 2.1	The system should show users who else from their network has consumed the same content item		
	Reader's App (ATC)	Show users from network	In -> UI Out -> Users



	Twitter Analytics (ATC)	Provide network information of a user	In -> User social id Out -> User network
	User Modelling (LiveTech)	Define network similarity metrics based on news consumption	In -> user news consumption Out -> network similarity metric
	Recommender (LiveTech)	Provide content based on network similarity	In -> Network similarity metric Out -> News selection
AF 2.2	The system should show users what else their network has shown, if there are differences		
	Reader's App (ATC)	Show more content	In -> UI Out -> News
	Recommender (LiveTech)	Provide content based on network similarities	In -> Criteria Out -> News selection
AF 2.3	The system should be able to show users the content item from another user (anonymously)		
	Reader's App (ATC)	Show content from another user	In -> UI Out -> News
	Recommender (LiveTech)	Provide content from another user	In -> User Out -> News selection
AF 3.4	The system should be able to offer both news content and entertainment		
	Recommender (LiveTech)	Offer both news content and entertainment, locally and globally relevant content.	In -> content request Out -> content
AF 3.5	The system should be able to offer both locally and globally relevant content		



	Recommender (LiveTech)	Offer both news content and entertainment, locally and globally relevant content.	In -> content request Out -> content
AF 3.7	The system should be able to give the user a timeline overview of events regarding a specific topic		
	Reader's App (ATC)	Display timeline view	In -> UI Out -> Timeline view
	Recommender (LiveTech)	Track events for a specific topic	In -> Events for a topic Out -> News selection
AF 3.8	The system should allow users to filter content by language		
	Reader's App (ATC)	Allow users to filter content by language.	In -> UI Out -> filtered content
AF 4.2	The system should allow for additional content sources, outside the consortium		
	Reader's App (ATC)	Allow for additional content sources, outside the consortium.	In -> content Out -> UI
	Producer's App (ENG)	Allow for additional content sources, outside the consortium	In -> outside sources Out -> CPN content
AF 5.1	The system must offer the user an easy to access and easy to understand overview of their profile		
	Reader's App (ATC)	Show profile info overview	In -> UI Out -> Profile info
AF 5.2	The system must offer users easy access to their profile in order to change settings and data		
	Reader's App (ATC)	Edit profile settings	In -> UI Out -> Profile settings



AF 6.1	The system must allow users to access content again that they have already opened before		
	Reader's App (ATC)	Mark already read content	In -> UI Out -> Content
AF 6.2	The system should allow users to consume content beyond their predefined timeframe after an interaction with the user (talkback)		
	Reader's App (ATC)	Ask user to show content beyond timeframe	In -> UI Out -> Content
	Recommender (LiveTech)	Provide content based on timeframe	In -> Criteria Out -> News selection
AF 6.3	The system should allow users to actively save articles for later consumption		
	Reader's App (ATC)	Ask user to show content beyond timeframe	In -> UI Out -> Content
AF 8.1	The system should allow users to search for specific topics they are temporarily interested in		
	Reader's App (ATC)	Provide search user interface	In -> UI Out -> Content
	Producer's App (ENG)	Index articles to allow search	In -> keywords Out -> Indexed document
	Recommender (LiveTech)	Provide search API	In -> Search criteria Out -> News selection
AF 8.2	The system should allow users to add this search as a temporary personalisation category		
	Reader's App (ATC)	Allow users to save searches and use for recommendation	In -> UI Out -> Saved searches



	Recommender (LiveTech)	Include search as category	In -> Category Out -> News selection
AF 8.3	The system should allow users to define a specific time frame for this temporary change		
	Reader's App (ATC)	Timeframe forms	In -> UI Out -> Saved timeframe
	Recommender (LiveTech)	Update category with timeframe	In -> Timeframe of category Out -> News selection
AF 9.1	The system should allow users to define keywords and logical combinations of them to exclude content from their personalisation		
	Reader's App (ATC)	Add keywords to search forms	In -> UI Out -> Search criteria
	Recommender (LiveTech)	Filter content by criteria	In -> Search criteria Out -> News selection
AF 9.2	The system should allow users to define a time frame per keyword/logical combination		
	Reader's App (ATC)	Add timeframe per keyword to search forms	In -> UI Out -> Search criteria
	Recommender (LiveTech)	Filter content by criteria	In -> Search criteria Out -> News selection
AF 9.3	The system should be able to overwrite this exclusion for important breaking rules		
	Reader's App (ATC)	Create form with overwrite field.	In -> UI Out -> Search criteria
	Recommender (LiveTech)	Filter content by criteria	In -> Search criteria Out -> News selection



PS 1.1	The system should show the access to items through users by numbers (who, when, how long)		
	Reader's App (ATC)	Provide usage statistics to the user	In -> UI Out -> Statistics
PS 1.3	The system should show which topics were most interesting to users		
	Reader's App (ATC)	Track information about most interesting articles and show topics	In -> UI Out -> Saved usage information
	User Modelling (LiveTech)	Define interesting metric	In -> topics Out -> user model update
	Recommender (LiveTech)	Provide most interesting topics	In -> User Out -> News selection

Table 6: Pilot 2 – Mapping between user requirements and functionalities

3.3 PILOT 3

3.3.1 Description and goals

The third pilot aims to demonstrate the overall CPN functionalities including all component features and evaluate the user experiences, the recommendations, the GDPR compliance, the full production side analytics and the licensing schema.

During this pilot, the final version of mobile CPN application will be released in order to gather detailed user feedback on all the CPN features and components.

3.3.2 Mapping

UR	Technology (Partner)	Brick	Functionality	Input/ Output
UP 1.3	The system should be able to offer personalised content on the basis of the users mood or values			
	User Modelling (LiveTech)		Define user mood related metrics	In -> User mood metrics Out -> user model update
	Recommender (LiveTech)		Provide users mood related criteria	In -> Criteria Out -> News selection
UP 4.1	The system must allow the user to choose preferred types of content			
	Reader's App (ATC)		Provide form to set preferences	In -> UI Out -> Preferred types information
	User Modelling (LiveTech)		Update user model based on preferred types	In -> Preferred types information Out -> User model update
UP 4.2	The system should set/refine preferred types of content based on the user's consumption habits and the timing			
	Reader's App (ATC)		Track user consumption habits and timing	In -> UI Out -> Usage information
	User Modelling (LiveTech)		Update user model based on usage information	In -> Usage information Out -> User model update
UP 4.3	The system should refine the user's preferred types of content through frequent interaction with the user (talkback)			
	Reader's App (ATC)		Show questions to the user	In -> UI Out -> Types of content



	User Modelling (LiveTech)	Update user model based on types of content	In -> Types of content Out -> User model update
UP 6.4	The system should be able to offer insights and advice based on what it learned about what a user consumed in relation to a certain entity (e.g. a place)		
	Reader's App (ATC)	Show questions to the user	In -> UI Out -> Types of content
	User Modelling (LiveTech)	Update user model based on types of content	In -> Types of content Out -> User model update
UP 6.5	The system should allow the user to delete part of the systems knowledge for specific time frames back in time from the moment of viewing		
	Reader's App (ATC)	Provide history information on system knowledge	In -> UI Out -> Remove systems knowledge
	User Modelling (LiveTech)	Update user model based on deletions of system knowledge	In -> Deleted system knowledge Out -> User model update
UP 7.1	The system should check on what device the user is consuming the content		
	Reader's App (ATC)	Save user device info	In -> UI Out -> Saved device info
	User Modelling (LiveTech)	Update user model based on usage device info	In -> Usage device info Out -> User model update
UP 7.2	The system should adjust its content offering based on the type of device the user is using		
	Recommender (LiveTech)	Provide content based on type of device	In -> Device type Out -> News selection



UP 7.3	The system should try to make smart use of device data to determine the surroundings of the user and adjust the content strategy accordingly		
	Reader's App (ATC)	Track device data	In -> UI Out -> Device data
	User Modelling (LiveTech)	Update user model based on device data	In -> Device data Out -> User model update
	Recommender (LiveTech)	Provide content based on device data	In -> Device data Out -> News selection
UP 8.1	The system should combine reading habits and knowledge about the user to provide smart updates on things the user could be interested in, even if this doesn't fit his/her set interests		
	Reader's App (ATC)	Display news that might interest the user	In -> UI Out -> Notification about news
	Recommender (LiveTech)	Implement news proposal algorithms	In -> User model Out -> News selection
UP 8.3	The system should be able to surprise the user with content, he/she would not have chosen themselves		
	Reader's App (ATC)	Display news that might surprise the user	In -> UI Out -> Notification about news
	Recommender (LiveTech)	Implement news "surprise" algorithms	In -> User model Out -> News selection
UP 9.6	The system should allow the user to add external data to update their profile		
	Reader's App (ATC)	Allow user to upload a file	In -> UI Out -> Uploaded data



	User Modelling (LiveTech)	Update user model based on external data	In -> External data Out -> User model update
AF 1.1	The system should offer users an overview of other sources, covering the same topic		
	Reader's App (ATC)	Show overview of other sources	In -> UI Out -> Overview
	Twitter Analytics	Get overview from TruthNest	In -> User social id Out -> Related sources information
AF 1.3	The system should offer the user an easy overview of what content from which sources he has consumed over a certain period of time		
	Reader's App (ATC)	Provide content usage statistics	In -> UI Out -> Usage statistics
AF 3.2	The system should offer the user a short overview of all important headlines at a specific point in time with access to more details upon request		
	Reader's App (ATC)	Display important headlines	In -> UI Out -> Important headlines
	Producer's App (ENG)	Tag headline as important	In -> outside sources Out -> CPN (tagged) content
	Recommender (LiveTech)	Implement "important headlines" algorithm	In -> User model Out -> Important headlines
AF 5.3	The system must make it transparent to the users why they are shown certain content, based on an item level		
	Reader's App (ATC)	Display justification based on item level	In -> UI Out -> Item level justification



	Recommender (LiveTech)	Provide justification based on item level	In -> Item Out -> Item level justification
AF 6.4	The system should be able to memorize where a user left off and restart at the same point		
	Reader's App (ATC)	Track left off/ start point	In -> UI Out -> Restarting point
AF 7.1	The system should offer user feedback requests in a playful/entertaining way		
	Reader's App (ATC)	Offer feedback requests in a playful way	In -> UI Out -> Feedback requests
AF 7.3	The system should allow users to assign both existing or new attributes (categories, moods etc.) to a content item		
	Reader's App (ATC)	Assign attributes to a content item	In -> UI Out -> Attributes to item
	Recommender (LiveTech)	Save attributes assigned to content item	In -> Item Out -> Saved attributes
AF 7.4	The system should be able to offer a feedback interaction to determine the ground level of personalisation based on mood, time and interest		
	Reader's App (ATC)	Display feedback information	In -> UI Out -> Feedback information
	Recommender (LiveTech)	Provide feedback interaction service	In -> Item Out -> Feedback information
PS 1.4	The system should be able to show these numbers during the creation process of the content		



	Reader's App (ATC)	Show feedback information	In -> UI Out -> Feedback information
	Producer's App (ENG)	Annotate content based on sentiment	In -> outside sources Out -> CPN content
	Recommender (LiveTech)	Provide feedback interaction service	In -> Item Out -> Feedback information
PS 2.1	The system should allow for an easy integration into the producers workflow		
	Producer's App (ENG)	Allow easy integration into the producers workflow	In -> outside sources Out -> CPN content
PS 2.2	The system should provide contract templates to allow freelancers to easily work together and with editors, to define and track the scope of individual contributions and expected revenues		
	Reader's App (ATC)	Provide contract templates	In -> UI Out -> Contract templates information
	Reward Framework (DCAT)	Provide rewards for contributors	In -> Contribution data Out -> Rewards
PS 2.3	The system should allow producers to transparently see how often their contributions are used and distributed to readers		
	Reader's App (ATC)	Provide producer's usage information	In -> UI Out -> Usage information
	Reward Framework (DCAT)	Provide usage statistics	In -> Producer id Out -> Usage information
PS 2.4	The system should allow producers to export the record of their publications through standardized and interoperable formats		



	Reader's App (ATC)	Provide export forms	In -> UI Out -> Exported data
	Producer's App (ENG)	Provide export publications service	In -> Producer id Out -> Usage information
PS 2.5	The system should allow for an easy contribution of content from different publishers through standardised interfaces		
	Reader's App (ATC)	Provide input forms	In -> UI Out -> Input forms
	Producer's App (ENG)	Provide import publications service	In -> Content Out -> Submitted content
PS 2.7	The system should allow editors to easily add missing attributes to articles manually		
	Reader's App (ATC)	Display attributes editing form	In -> UI Out -> Attributes editing
	Producer's App (ENG)	Provide edit attributes service	In -> Attributes Out -> Updated attributes

Table 7: Pilot 3 – Mapping between user requirements and functionalities

3.4 INNOVATIVE COMPONENTS ANALYSIS

During workshops, survey and interviews, as described on D1.1, as well as the requirements collection, a very important focus was given to innovative techniques and approaches.

In fact, in order to reach the next level of personalised and context aware recommendation, it is important to go beyond the state of the art and try to introduce novel techniques and features.

The collection of new ideas and concepts was reported in D1.3 Innovative CPN Components⁶, in which were identified four new innovative domains and related components that could bring the CPN solution to this next level of innovation.

Newsbots and smart speakers⁷



News bots, smart speakers and digital assistants will allow to make the user experience of news consumption as seamless as possible. They rapidly gain in popularity and offer a whole new paradigm for personalisation.

To develop a suitable solution for CPN, we will have to take into consideration the challenging aspects of text- and voice-based bot applications. Among them, the users' limited attention span and possibly quick frustration with chatbots, limitation of natural language processing technologies, as well as the complexity to accommodate highly individual language usage and language differences among our international audience (without making it too tiresome to learn to communicate with the system).

External and highly contextualized datasets⁸

External data sources will add intelligence to personalisation on an individual level, considering a user's context while increasing engagement.

In order to have a highly contextualised personalisation, it is important to learn the user habits and context, adding many data sources that infer our knowledge about him (e.g. geolocation, wearable and social media) and piecing together the information collected to infer context.

Security, trust, control and transparency⁹

Some user surveys conducted by media partners highlight that users are willing to share data with an application in order to get a more personalised offer but they want to be in control and to be aware of their data usage.

This implies that any new application or service, trying to serve a personalised offer of any kind, has to build up trust among its users from the first moment on. Looking at the current situation, we think there are three main aspects to take into consideration to build this level of trust:

- a) Security: The project must convince users that their data is safe from external access (data hacks) and not to be shared/sold to anyone without their explicit consent (data brokerage)
- b) Transparency: Users must always be able to see what data about them is stored and what it is used for, in an understandable way.
- c) Control: Users must also be able to make changes to their data, e.g. withdraw their consent for its use, change it or even delete data, in parts or completely in an easy fashion.

Multi-layered personalisation¹⁰

⁶https://www.projectcpn.eu/s/CPN_D13_Innovative-CPN-Components_20180228_V10-mngx.pdf

⁷ D1.3 – CPN Innovative components (Chapter 2.1)

⁸ D1.3 – CPN Innovative components (Chapter 2.2)

⁹ D1.3 – CPN Innovative components (Chapter 2.3)

¹⁰ D1.3 – CPN Innovative components (Chapter 2.4)



A personalisation based on multiple levels and using sophisticated techniques to take the personal traits of users into account will allow the system to create parallel article versions that will be better adapted to the individual user's characteristics instead of today's one-size fits all approach.

In the following paragraphs is described an analysis of these innovative components and in particular, how the CPN solution could approach the implementation of these components starting from the use cases already defined and the functionalities offered by the technology bricks expected by the CPN platform.

3.4.1 User Scenarios

Starting from the User scenarios described on D1.2, it was envisioned how the innovative components could be integrated into each of them.

Newsbots and smart speakers

User Scenario 1: The Power News User

Smart speakers could help the Power news users to make an effective use of personalized content.

External and highly contextualized datasets

User Scenario 2: The Data-Minded Sceptic

In this scenario, to avoid the fear of missing out and the overload of contents, it is necessary to have a contextualized and personalized dataset but also give the possibility to the user to increase the data sources (e.g. adding data from social media).

User Scenario 3: Light News User

The Light news user is an infrequent user that usually prioritise content from a single source, thus it is important to give him a highly contextualized dataset.

User Scenario 4: The Product Manager's View

An innovative analytics feature allows the editors to have a better perception on the use of content by users, helping them to produce better content and identify new trends.

User Scenario 5: The freelancer's view

Content analytics allows freelancers to have the complete control over the contents, including payment systems.

Security, trust, control and transparency

User Scenario 2: The Data-Minded Sceptic

In this scenario, it is very important to give to the user a complete control of its personal information in a transparent way, thus creating a sense of trust that allows him to share more information necessary for personalization.

User Scenario 4: The Product Manager's View

With the introduction of GDPR, it becomes much more complicated to manage user's data. A component that ensures data security and transparency automatically, simplifies the management of all these aspects to the Product Manager.

User Scenario 5: The freelancer's view

An automated and transparent component that handles the user data and the contents, provides a trusted access for the freelancer that can understand the content license without



legal experts and tracks all the feedback related to its contents, which are recorded in a transparent way.

Multi-layered personalisation

User Scenario 1: The Power News User

The Power news user wants to change his personalisation settings in a deep way, thus obtaining a very personalized level of recommendation, suitable for his needs.

User Scenario 3: Light News Use

It is important to have a system that identifies the Light news user, in order to give him a very innovative personalisation and convince him of the effectiveness of the CPN application.

User Scenario 4: The Product Manager's View

Innovative mechanisms that trace the user consumption habits and identify new user groups, allow the Product Manager to better identify the user needs and to adapt their contents accordingly.

3.4.2 Technology Bricks

Starting from the description of technology bricks as reported on D3.1, a set of functionalities that could help to implement these new innovative components were highlighted.

These functionalities can be exploited to define new modules or improve the existing ones, in order to achieve innovation. In fact, the CPN platform, as described in the D2.1 CPN Reference Architecture, allows by design to improve the modules deployed or to combine some functionalities to create new ones.

Newsbots and smart speakers

Relation Extractor

It offers some standard NLP functionalities that could be exploited to add a more human touch. In fact, with NLP we could be able to “train” the newsbot on the various interactions it will go through, and help streamline the responses it outputs.

External and highly contextualized datasets

Producer's App

It already includes integration with user personal social networks and the possibility to integrate other personalized external sources (RSS feed, Meteo and transportation API. etc...). It also offers a basic analytics feature.

Twitter analytics

It includes Twitter integration and analytics information that can be used to reveal user habits.

Sentiment and Topic extractor

They can analyse content extracted from external sources and extrapolate further information about the users.

Security, trust, control and transparency

Personal Data Receipts



CPN has been designed to be GDPR compliant and to ensure security, trust, control and transparency for its users. This module provides a human-readable record summarising in a simple and clear way what personal data an organisation is collecting about an individual, for what purpose, how they are stored, for how long and if any third party sharing is allowed.

Recommender

It ensures transparency during recommendation process, because it provides insights about each recommendation.

Multi-layered personalisation**User modelling**

It offers a basic user-profiling feature, collecting user consumption habits and metadata as location. It also extracts topics of interests directly from news articles

Sentiment

It is a trainable classification system, which offers polarity detection, complex opinion and sentiment extraction, spam, hate-speech and flame detection. It extracts additional meta-data information from news content.

Recommender

It offers semantic extraction features and combines the results from user modelling and sentiment modules.



4 PLATFORM REQUIREMENTS

The purpose of the platform requirements is to specify the overall solution requirements that will govern the development and implementation of the platform. It will also establish security, scalability, capacity and architecture requirements.

4.1 CPN PLATFORM

The CPN platform is an Open, Virtual platform based on a microservices architecture that has been described in the D2.1 Reference Architecture.

The figure below represents the basic architecture:

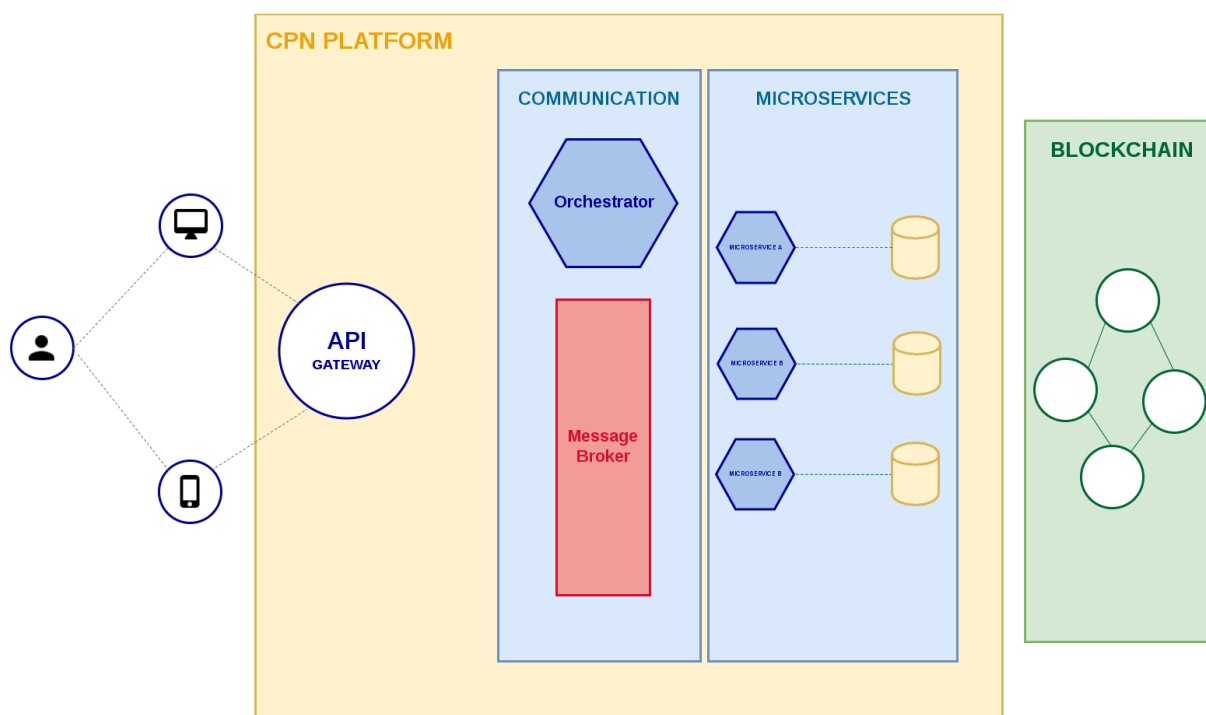


Figure 1: CPN platform architecture

The CPN platform includes, in addition to the Technology Bricks which are implemented as microservices and deployed in the platform, a series of core components that allow communication with external components (API gateway) and between internal components, both synchronous (Orchestrator) and asynchronous (Message Broker).

The architecture requirements was already detailed in D2.1 CPN Reference Architecture, thus in this chapter are described the technologies and the external components choices to satisfy these requirements.

4.1.1 Container management platform

For the management of the platform, it was decided to use a container management, which would allow easily configuring of the platform, managing the deployment of all the microservices and monitoring their status. This container management platform is Rancher v1.6¹¹.

Rancher is an open source software platform that enables organizations to run and manage Docker container in production environment.

Rancher takes in raw computing resources from any public or private cloud in the form of Linux hosts. Each Linux host can be a virtual machine or physical machine. Rancher does not expect more from each host than CPU, memory, local disk storage, and network connectivity

Rancher implements a portable layer of infrastructure services designed specifically to power containerized applications. Rancher infrastructure services include networking, storage, load balancer, DNS, and security. Rancher infrastructure services are typically deployed as containers themselves, so that the same Rancher infrastructure service can run on any Linux hosts from any cloud.

Rancher includes a distribution of all popular container orchestration and scheduling frameworks today, including Docker Swarm, Kubernetes, and Mesos. In addition to these, it supports its own container orchestration and scheduling framework called Cattle.

Cattle¹² is the orchestration framework chosen for the deployment of the first version of CPN platform. This framework organizes the modules into stack (a group of services) and services.

There are two categories of stack:















- ➔ User stacks: include all the stacks deployed by the user
- ➔ Infrastructure stacks: include all the default stacks offered by Cattle framework

¹¹ <https://rancher.com/docs/rancher/v1.6/en/>

¹² <https://rancher.com/docs/rancher/v1.6/en/cattle/stacks/>



The figure below shows the list of stacks deployed for the first prototype on CPN host.

User Stacks					Sort By: State Name	
 + api-gateway	<input type="button" value="Add Service"/>	1 Service	1 Container			
 + ds4biz	<input type="button" value="Add Service"/>	6 Services	6 Containers			
 + IMEC-relation-extractor	<input type="button" value="Add Service"/>	2 Services	2 Containers			
 + kafka2	<input type="button" value="Add Service"/>	1 Service	1 Container			
 + personal-data-receipts	<input type="button" value="Add Service"/>	1 Service	1 Container			
 + producer-app	<input type="button" value="Up to date"/> <input type="button" value="Add Service"/>	2 Services	2 Containers			
 + Zookeeper	<input type="button" value="Add Service"/>	1 Service	1 Container			

4.1.2 Core components

Even if the CPN Platform is open and extensible and the modules installed may be replaced or improved according to customer needs, some core components are necessary to have a basic configuration and deployment.

As described into the D2.1 CPN Reference Architecture, the CPN microservices architecture implies the application of API Gateway pattern and two kind of communication systems: messaging and orchestration. In order to satisfy these architectural requirements three core components have been deployed within the platform:

- ➔ API Gateway
- ➔ Orchestrator
- ➔ Message Broker

API Gateway

The API Gateway represents the access door for external application that want to exploit CPN innovative services.

This component serve all different client applications (mobile, web, etc.) and centralize some middleware functionalities as authentication, logging, security, etc.

In CPN platform a microservices API gateway, based on Express.js framework has been deployed, namely the Express Gateway module.

Express Gateway¹³ is open source and distributed under Apache 2.0 License¹⁴ and it was chosen for its ease of use and flexibility and in particular for the following benefits:

- ➔ Configurable via YAML
- ➔ Language agnostic
- ➔ Extensible with any Express.js middleware
- ➔ Run anywhere with Docker
- ➔ Many authentication and authorization method supported (e.g. JWT)

¹³ <https://www.express-gateway.io/>

¹⁴ <https://github.com/moby/moby/blob/master/LICENSE>



The CPN API Gateway was extended with a GUI that permit to view all documented APIs and test them directly via browser.

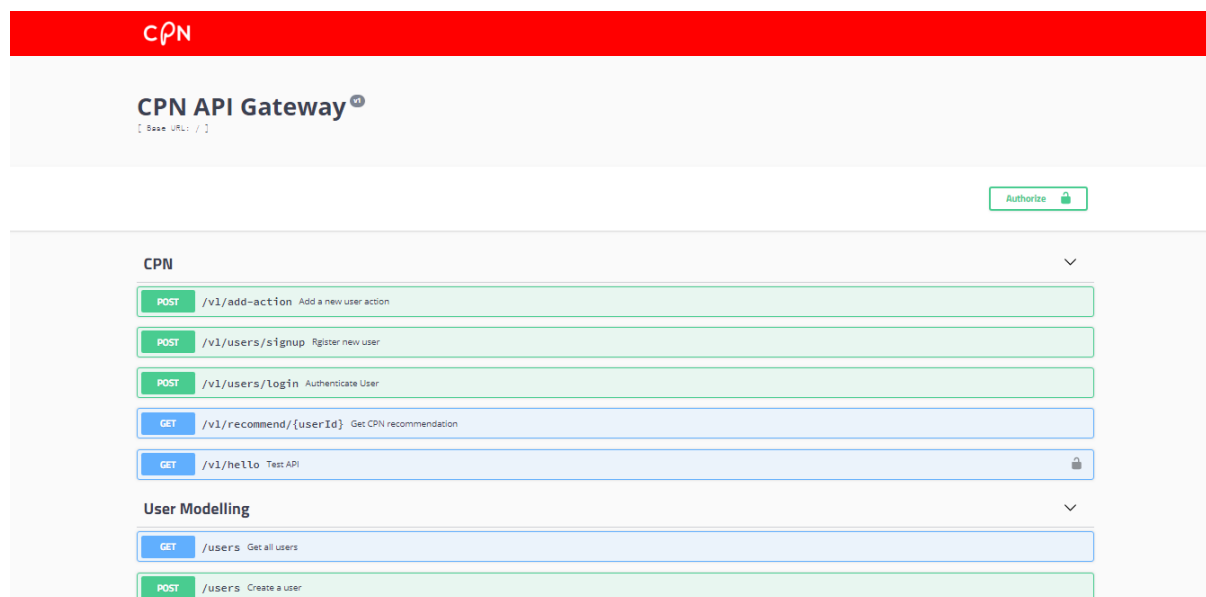


Figure 2: CPN API Gateway GUI

All the APIs exposed on the CPN API gateway must be documented in a standard API documentation format and in particular in a YAML¹⁵ file with Open API specification Swagger v2.0¹⁶.

Thanks to this standardization all the APIs are human readable and testable directly through the API gateway GUI.

Orchestrator

Orchestration is the traditional way of handling interactions between different services in a service-oriented architecture. With orchestration, there is typically one controller that acts as the “orchestrator” of the overall service interactions.

We developed a customized orchestrator module and deployed it as Docker container, to get the following benefits:

- ➔ unbundling between gateway and orchestration functionalities
- ➔ no violation of the single responsibility principles
- ➔ more flexibility to implement new processes and scaling APIs

¹⁵ <https://en.wikipedia.org/wiki/YAML>

¹⁶ <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

¹⁷ <https://swagger.io/>

The orchestration module covers all the synchronous processes inside the platform, exploiting the APIs exposed by technology bricks. In particular, a new component based on Node.js technology was implemented and deployed as a microservice within the platform.

Implementing the orchestrator in a separate microservice brings the following benefits:

- ➔ unbundling between gateway and orchestration functionalities
- ➔ non-violation of the single responsibility principle
- ➔ more flexibility to implement new processes and scaling APIs

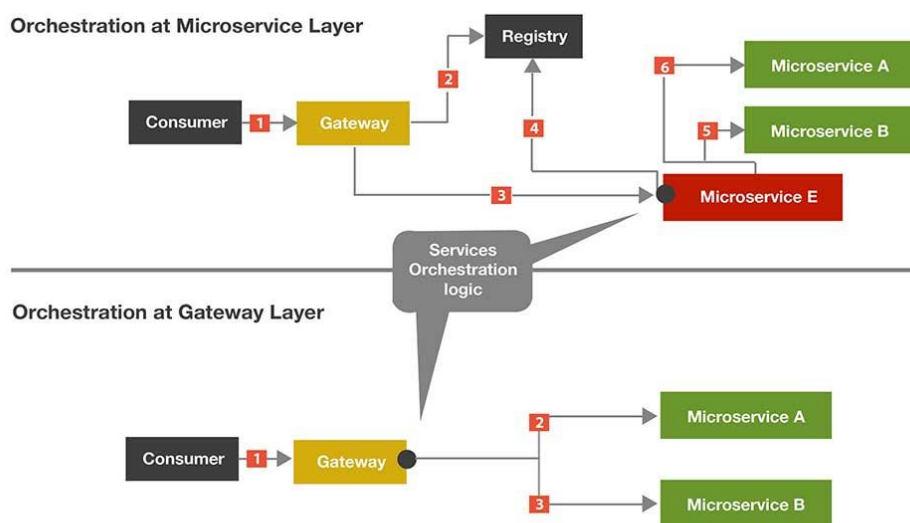


Figure 3: Two ways for implementing orchestration¹⁸

Message Broker

¹⁸ https://www.globallogic.com/gl_news/microservice-architecture-api-gateway-considerations/

The messaging pattern allow the microservices to collaborate minimizing their coupling and improving the flexibility of the platform.

To implement this pattern a message broker is needed. A message broker (or queue manager) is a software where queues can be defined, applications may connect to the queue and transfer a message onto it.

In CPN platform, Apache Kafka¹⁹ and Zookeeper²⁰ have been chosen to allow messaging and asynchronous communication among microservices. They are both open sources and distributed under Apache 2.0 License.

Apache Kafka is a distributed streaming platform that allow publishing and subscribing streams of records, very similar to a standard message queue. Kafka is deployable as a cluster of one or more servers that stores stream records in categories called topics. Each record consists of a key, a value and a timestamp.

Zookeeper is a software tool for distributed services coordination. In particular, it is a service for maintaining configuration information, naming, providing distributed synchronization, and providing group services.

Both the modules were deployed into the CPN platform as Docker containers and are available to all the other modules.

The technology bricks that use asynchronous communication must implement the rule of integration and setup as environment variable the local address of Kafka.

4.1.3 CPN Catalog and Registry

In order to makes available the modules developed by the partners, we need a private container registry.

CPN implements a private container registry and uses it as a repository for technology bricks and innovative components but also for new modules and components. CPN users can load their own modules and make them available to the CPN platform in an automated and secure way.

The CPN platform also includes a GUI to display all the modules and components available on the CPN repository, named CPN catalog. This catalog is a sort of marketplace for CPN modules, provided with description, technical documentation and usage guidelines, from which the user can choose the modules to be installed into the platform.

The figure below shows the GUI with a catalog example:

¹⁹ <https://kafka.apache.org/>

²⁰ <https://zookeeper.apache.org/>



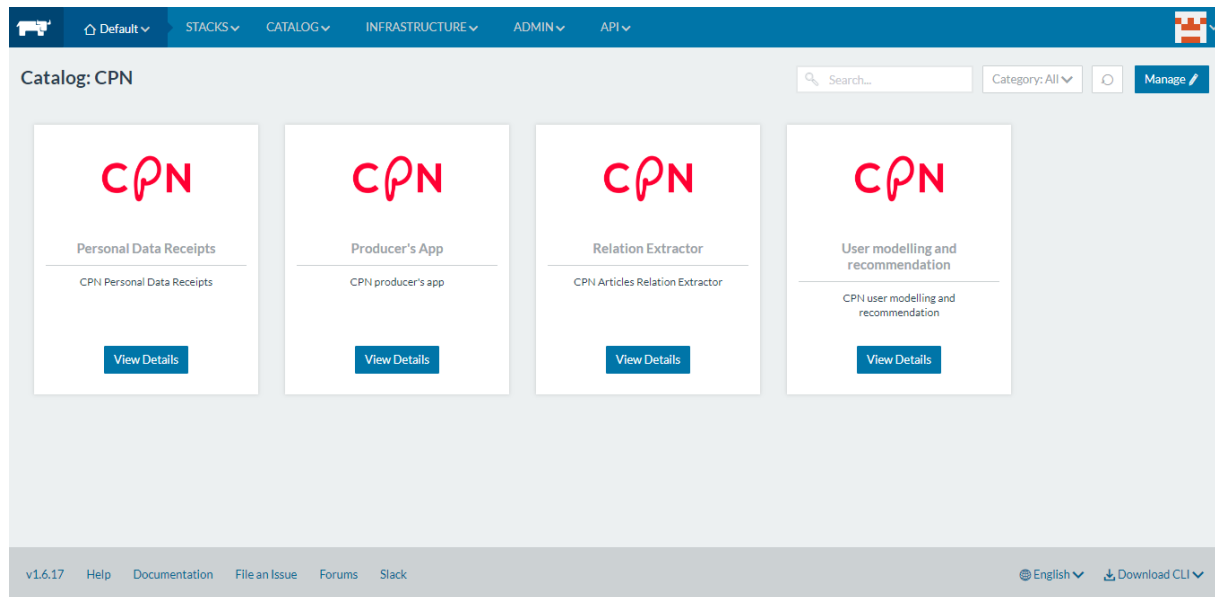


Figure 4: CPN catalog example

4.2 FUNCTIONAL REQUIREMENTS

Following the analysis carried out and after the architecture definition of the CPN platform with its related components, two further actors were identified within the CPN ecosystem: the platform administrator and the developer.

The platform administrator is the one who actually manages the platform from an administrative and technical point of view and monitors its status.

The developer is the one who exploits the services offered by the platform to create innovative client applications. It also has the possibility to create new services and to integrate them into the CPN platform by exploiting or improving existing ones.

For the scenarios involving these actors, some functional requirements have been defined for the CPN platform:

ID	Description
PR-1	The Platform should have a User friendly UI in order to manage the platform components and host
PR-2	The Platform UI should allow creating and managing different User roles with different permissions (at least two: administrator and developer)
PR-3	The Platform UI should contain a section with CPN components catalog
PR-4	The Platform UI should have a section with the API documentation
PR-5	The administrator should be able to monitor the entire platform resources
PR-6	The administrator should be able to manage each single component of the platform
PR-7	The administrator should be able to add new component into the platform from a catalog
PR-8	The developer should be able to add a new component into the CPN catalog
PR-9	The developer should be able to view and test all the existing services
PR-10	The developer should be able to use the existing services

Table 8: CPN platform functional requirements

4.3 NON-FUNCTIONAL REQUIREMENTS

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. Non-functional requirements complement the functional requirements of a system and are often a result of the analysis of the availability, stability, scalability, etc. but also other aspects such as e.g., safety, security & privacy and performance.

ID	Description
NFR-1	The system should be able to scale horizontally or vertically depending on the demands related to data ingestion, processing and storage.
NFR-2	It must be possible to decompose the solution in different microservices
NFR-3	The system should enable a versioned and automated deploy mechanism for the microservices
NFR-4	The system should include a centralized authentication and authorization mechanism
NFR-5	The system should enable monitoring and logging mechanisms
NFR-6	The system should enable synchronous communication among the microservices, through orchestration mechanisms
NFR-7	The system should enable asynchronous communication among microservices, through messaging mechanisms
NFR-8	The system should offer optimized endpoints in base of different kind of client's applications (mobile, web, smart tv...)
NFR-9	The system should be extensible with new functionalities and new components
NFR-10	The APIs exposed by the microservices should be documented in a standard format.
NFR-11	The system should keep the data private and only accessible via authenticated APIs
NFR-12	The data produced, consumed and transformed shall be documented in an information model which shall also include the relationships between information types.
NFR-13	The system shall be able to exchange data with a great number of devices and, at the same time, preserving its computational capacity.



Table 9: CPN Platform non-functional requirements



5 APPROACH AND METHODOLOGY

As already mentioned in the chapter 3, the CPN project followed an iterative incremental approach for the platform development, based on the SCRUM Agile methodology.

It is based on three phases, each of which consists of a full development cycle and provides the release of the project platform and the related components (technology bricks and core components) in a prototype. This prototype is evaluated adopting the living-lab methodology. Effective user research and feedback is collected and being used as a base of the next phase.²¹

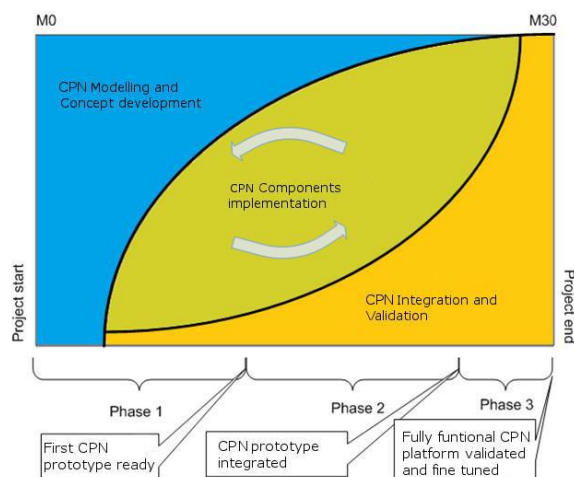


Figure 5: CPN implementation approach

More specifically, CPN adopts a longitudinal research design (with inter-case benchmark comparing cases) in the three selected countries: Belgium, Germany and Cyprus. This is described in the figure below:

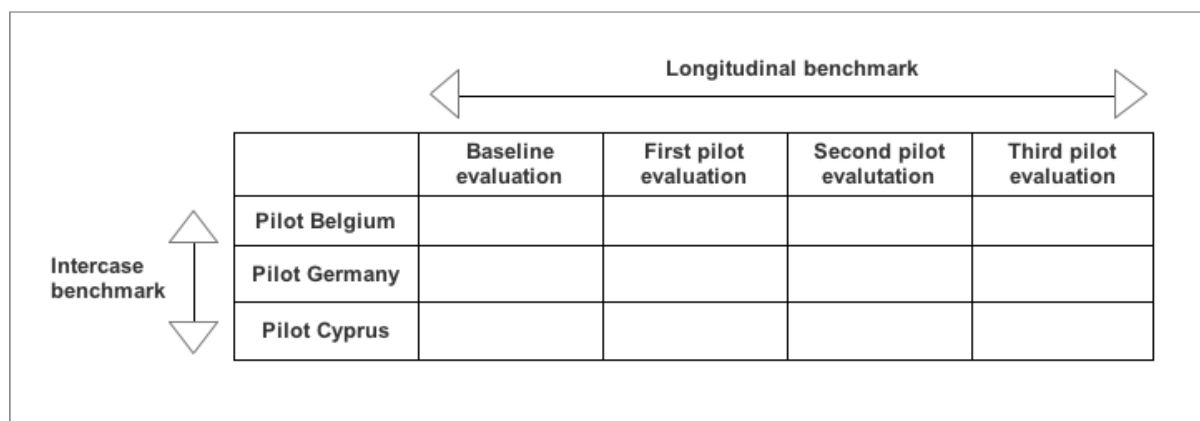


Figure 6: Longitudinal research design with inter-case benchmark

To be more precise, the evaluation activities will be described in D4.2 Cycle 1 piloting report, which is expected at the end of the first pilot execution at October 31, 2018.

From a technical point of view, as already mentioned, a SCRUM approach was followed.

In particular, a 3-steps process was defined: two starting activities steps and one cyclic step for task development.

- ➔ Mapping Requirements/Technology Bricks
- ➔ Requirements prioritization
- ➔ Sprints

This process is already in progress in the first cycle of the implementation and will be provided in the first version of prototype in early September.

Mapping Requirements/Technology Bricks

This activity, reported in this document, was made in collaboration between the media partners and the technical partners. Its primary target was to evaluate the technology level offered by technology bricks in relation to the requirements collected. Furthermore, the focus was to allow an effective evaluation of the technical work plan and schedule the activities.

This activity was carried out for all the requirements, including those of the second and third prototypes, laying the foundations for the subsequent implementation cycles. However, since the project is based on agile methodologies, this does not imply that the mapping of these features cannot be **reviewed and refined at the beginning of each phase**, based on feedback gathered in the previous phases and on the progress of the platform and technology bricks.

Requirements prioritization

This activity, which will be described in detail in D4.2 Cycle 1 piloting report, was carried out only on first prototype requirements and will be repeated in the subsequent phases.

The media partners (VRT, DW, DIAS), which in the case of the Scrum methodology assumed the role of Product Owners, assigned a priority for each user requirement. The result of this activity was an ordered list of requirements, named Prioritized Backlog.

Sprints

Sprints are the base unit of the SCRUM framework development activities. They are of fixed duration and are repeated throughout the duration of the project.

In the CPN technical work plan, each sprint duration was of two weeks and each sprint consisted of:

- ➔ Sprint planning
- ➔ Sprint execution
- ➔ Sprint review

²¹ As described in CPN Grant agreement 1.3.2.3.2 pag.136



The sprint planning is the starting phase of a sprint. During this activity, the tasks to be completed in the sprint are selected.

The sprint execution is the second phase, which continues for all the duration of the sprint (two weeks). During this phase, the tasks are executed and completed.

The sprint review is the final phase of the sprint. In this phase, the completed and pending tasks are being discussed. The completed tasks are demonstrated by the Scrum Team. The Scrum Team also discusses what went well during the Sprint, what problems it ran into, and how those problems were solved. Finally, the sprint review provides valuable input to subsequent sprint planning.

The tool selected to follow the implementation activity is Trello.²² It is a web-based project management application, allowing managing of all the activities in an agile way. It consists of a “board” where you can define a list of activities to do (in CPN, the product backlog), order them in base of your prioritization and assign to other members.

In the figure below, the Trello board used for CPN implementation phase is shown:

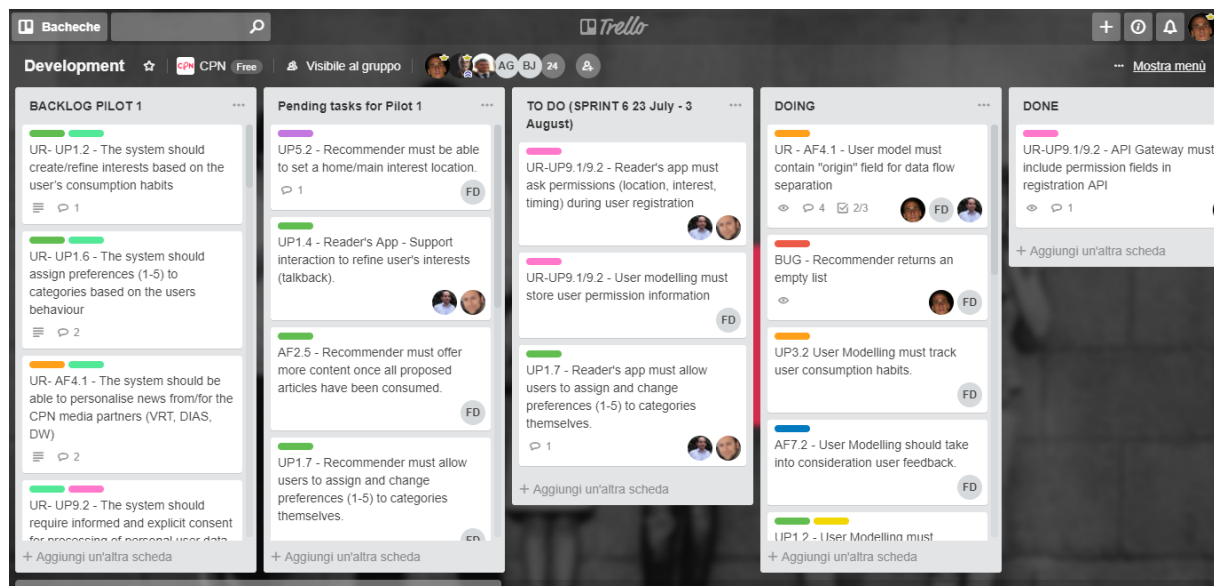


Figure 7: CPN Trello board

²² <https://trello.com/>

6 CONCLUSIONS

The D1.4 CPN technical requirements includes all the requirements gathered during other WP1 tasks and integrates them with the platform functional and non-functional requirements.

In addition, it offers an overview of all the activities carried out so far and collected in other technical deliverables, such as the D2.1 CPN Reference architecture and the D3.1 Initial Design and APIs of Technology Bricks. Furthermore, it defines a precise technical work plan for the development of the platform and its components, as long as their integration.

Particular attention was paid to the mapping between the Technology Bricks and the requirements but also to the possibility of evolving the CPN platform with the innovative components identified in the D1.3 Innovative components.

The activities described in this document will be used to define the releases processes of the CPN solution and especially the platform and technology bricks evolution. This release process aims to a final release with all the requirements satisfied. Following a cyclical approach, it can be modified or integrated in the course of the project. In fact, any feedback from phase 1, which is currently underway at the time this document is written, will be used to guide the way forward for the next steps.

