Grant Agreement No.: 761488

D2.3: CPN Open Virtual Platform v2

CPN Platform v2 - Accompanying Report

Work package	WP 2
Task	T2.1 – T2.2 – T2.3 – T2.4
Due date	31/05/2019
Submission date	
Deliverable lead	ENG
Version	1.0
Authors	Ferdinando Bosco (ENG), Vincenzo Croce (ENG)
Reviewers	Fulvio D'Antonio (LiveTech)
Keywords	CPN – Open Virtual Platform – Microservices Architecture

Document Revision History

Version	Date	Description of change	List of contributor(s)
V0.1	06/05/2019	1st version of the deliverable with table of contents	Ferdinando Bosco (ENG) Vincenzo Croce (ENG)
V0.2	24/05/2019	Draft Version with contributions from partners	Ferdinando Bosco (ENG) Vincenzo Croce (ENG) Stamatis Rapanakis (ATC) Joris Mattheijssens (VRT) Steven Semples (VRT)
V0.3	27/05/2019	Revision of contributions and extension of the deliverable	Ferdinando Bosco (ENG) Vincenzo Croce (ENG) Fulvio D'Antonio (LiveTech)
V0.4	29/05/2019	Reviewed version	Fulvio D'antonio (LiveTech)
V1.0	30/05/2019	Final Version	Ferdinando Bosco (ENG) Vincenzo Croce (ENG)

Co-funded by the Horizon 2020 Framework Programme of the European Union

DISCLAIMER

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

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	

Co-funded by the Horizon 2020 Framework Programme of the European Union

EXECUTIVE SUMMARY

The CPN project foresees three major releases of the CPN Open Virtual platform that include functionalities aligned with the requirements gathered for the three CPN pilots. Each platform release includes specific functionalities, chosen after a process of evaluation and prioritization of the user requirements.

The second version of the CPN open virtual platform, delivered as evolution of the first version, will be described on this document. It is the CPN Open Virtual Platform v2 accompanying report which describes all the updates, and improvements of the platform respect on the previous version, described on D2.2 CPN Open Virtual Platform v1 [1]

In particular, the first chapter introduces the scope of the deliverable and the process of implementation and delivery of the CPN platform.

The second chapter describes all the updates and improvements of the platform respect on the previous version, focussing on the platform core components and the list of integrated technology bricks.

The third chapter will focus on the requirements satisfied, as defined in "D3.3 CPN Technology Bricks v2"[2], and the list of functionalities implemented. It will describe the results accomplished in terms of requirements satisfied and functionalities provided for the execution of the second pilot.

The fourth chapter describes the sidetracks activities conducted by the technical partners with two of the media partners to demonstrate and validate the efficiency, efficacy and interoperability of the platform. In particular, the recommender system provided by the CPN platform was integrated into two client applications: a mobile app and a smart-TV app.

[1] https://www.projectcpn.eu/s/CPN_D22_CPN_Open_Virtual_Platform_v1_20180629_v10.pdf

[2] https://www.projectcpn.eu/s/D33-Technology-Bricks-V2.pdf

Co-funded by the Horizon 2020 Framework Programme of the European Union

TABLE OF CONTENTS

Co-funded by the Horizon 2020 Framework Programme of the European Union

LIST OF FIGURES

Co-funded by the Horizon 2020 Framework Programme of the European Union

LIST OF TABLES

Co-funded by the Horizon 2020 Framework Programme of the European Union

ABBREVIATIONS

API	Application Programming Interface
ATC	Athens Technology Center
CPN	Content Personalisation Network
DCAT	Digital Catapult
DW	Deutsche Welle
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
RDF	Resource Description Framework
RSS	RDF Site summary
UI	User interface
UR	User requirement
VRT	Vlaamse Radio-en Televisieomroep
YAML	YAML Ain't Markup Language

Co-funded by the Horizon 2020 Framework Programme of the European Union

1 INTRODUCTION

The CPN project foresees three major releases of the CPN Open Virtual platform that include functionalities aligned with the requirements gathered for the three CPN pilots. Each platform release includes specific functionalities, chosen after a process of evaluation and prioritization of the user requirements.

The CPN Open Virtual Platform v2, described in this document, mainly focuses on the second pilot execution. It integrates all the new technology bricks delivered and implements a set of functionalities detailed on D1.4 Technical requirements (platform and service requirements)[3]. In particular, it focuses on the functionalities necessaries to satisfy the user requirements for this second pilot.

The process of prioritization of the requirements, already described on D2.2 CPN Open Virtual Platform[4], was performed to plan the activities and define the technical tasks necessaries to delivery both the technology bricks and the new version of the platform.

In addition, the second version of the CPN platform now stands in a more mature context and it is enriched with processes that improve reliability, safety and performance. In fact, the core components were improved and extended and integration tests were conducted in order to demonstrate the planform flexibility.

3 https://www.projectcpn.eu/s/CPN_D14_Technical_requirements_platform_and_service_requirements_20180830_v10.pdf

Co-funded by the Horizon 2020 Framework Programme of the European Union

2 PLATFORM IMPLEMENTATION – UPDATES

In this second version of the CPN Open Virtual Platform, new functionalities has been implemented and made available, through the integration of new modules and the evolution of existing ones, including the core components of the platform.

In order to have an overview of the status of the platform, below are reported all the updates of the core components and in particular of the API Gateway and the Orchestrator, as well as the list of the technology bricks integrated or updated in this version. The details of the features and improvements of technology bricks are instead reported in D3.3 CPN Technology Bricks v3.

2.1 CORE COMPONENTS

As already described into the D2.2 CPN Open Virtual Platform v1, three core components were deployed into the CPN platform to allow communication among the internal components and exploitation of the CPN functionalities:

- **API Gateway,** it is the entry point of the CPN platform. It allows accessing the CPN functionalities for both client applications and developers. It also handle authentication, authorization**,** logging and scalability
- **Orchestrator**, it handles the synchronous communication among the technology bricks
- **Message Broker,** it is based on Apache Kafka and it handles the asynchronous communication among the technology bricks

Co-funded by the Horizon 2020 Framework Programme of the European Union

2.1.1 Api Gateway

The API Gateway represents the access door for external client applications and developers that want to exploit CPN innovative services

It is based on Express.js framework[5] and configurable via YAML file. In particular, the YAML configuration file contains all the accessing and authorization rules and the mapping between external contexts and internal services.

The following table resume all the APIs exposed until now, their contexts and the internal mapping on the CPN platform:

Name	Context	Internal Mapping / Services
CPN	/v1	All the services for the client applications
CPN - Admin	/v1/admin	All the administration services
User Modelling	/users	User Modelling – user profiles services
	/topics	User Modelling – topic services
Recommender	/admin	A/B Recommender – recommender configuration services
	/recommend	Recommender – recommendation services
Personal Data Receipt	/pdr	Personal Data Receipt – PDR services
Producer's App	/producer	Producer's App – search contents and analytics services

Table 1: CPN APIs List

As shown on the figure below, the APIs was documented with the OpenAPI v2.0 standard [6]and available through the API gateway UI, based on Swagger[7].

[5] https://expressjs.com/

[6] https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md

[7] https://swagger.io/

Co-funded by the Horizon 2020 Framework Programme of the European Union

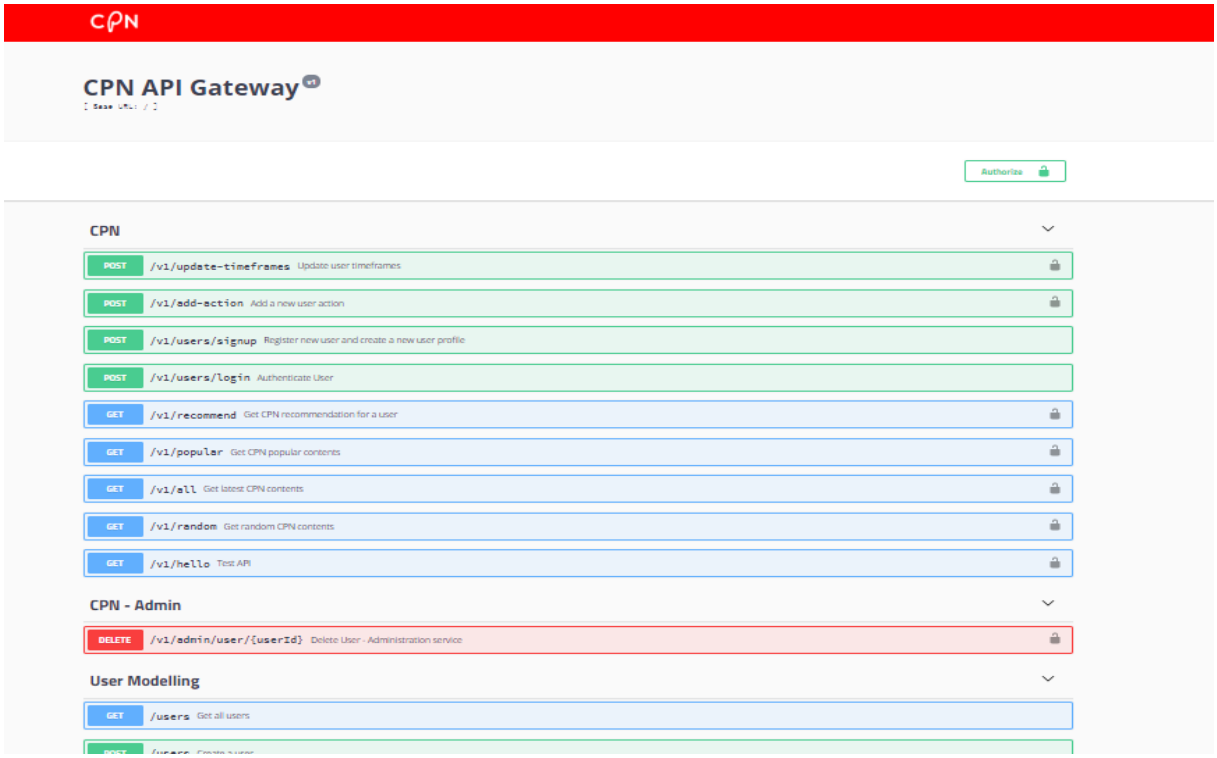


Figure 1: CPN API Gateway Web UI

In this version of the platform, a JWT mechanism was included for the authentication and an authorization system, based on roles was implemented.

The following roles have been defined:

- **CPN Administrator**, it is a super-user. It manages all the configuration of the CPN platform, can create new roles and users.
- **Media Administrator,** it manages all the data and components concerning a media company. It can access to media contents and configure the recommendation engine for a specific media company. In addition, can authorize developers to access to its own data.
- **Developer**, it can access to CPN API, both client applications and technology bricks. It is authorized by a media administrator, following an agreement.
- **User,** it does not access directly to the APIs but he is recognized by an authentication system and it has the total control to its own data.

Co-funded by the Horizon 2020 Framework Programme of the European Union

The combination of the JWT based authentication and the new authorization system has a high focus on security and GDPR compliance. in fact, the personal JWT ensure that each user can only access to its own data and at any time it can ask for retrieve or delete them. In addition, the media administrators can manage their own data and configure their personal recommender, without the risk of collisions with other tenants and can directly manage the agreement with developers to give them access to specific data and/or services.

2.1.2 Message Broker

The CPN Message Broker is based on Apache Kafka and Zookeeper. It is used by the technology bricks to access to CPN data in asynchronous way.

In particular, the following topics was defined:

- **Articles,** the list of the articles retrieved from media companies
- **Actions,** the list of user actions collected during the usage of client applications
- **Notifications,** the list of push notifications to send to users

Co-funded by the Horizon 2020 Framework Programme of the European Union

The following figure show how the CPN technology bricks exploit the message broker as publisher or subscriber:

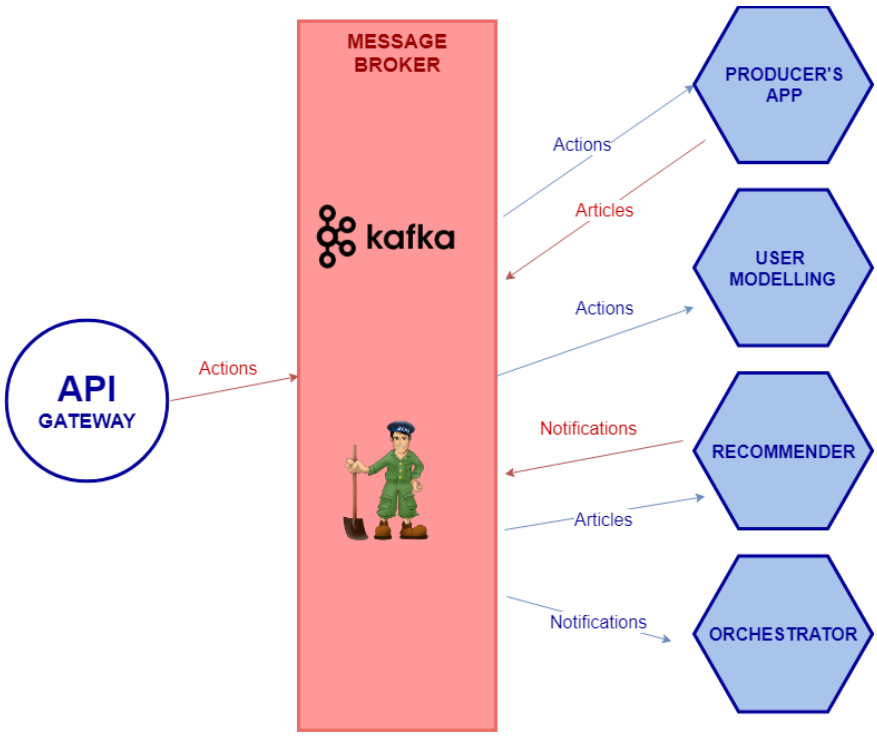


Figure 2: Message Broker integration

2.1.3 Orchestrator

The orchestration module cover all the processes inside the platform that need synchronous communication among technology bricks. It do it exploiting the APIs exposed by each technology brick.

In particular, the orchestrator, deployed as a microservice inside the CPN platform, interfaces with the following technology bricks, through configurable routes:

- **Recommender**
- **User Modelling**
- **Producer's App**
- **Message Broker**

Co-funded by the Horizon 2020 Framework Programme of the European Union

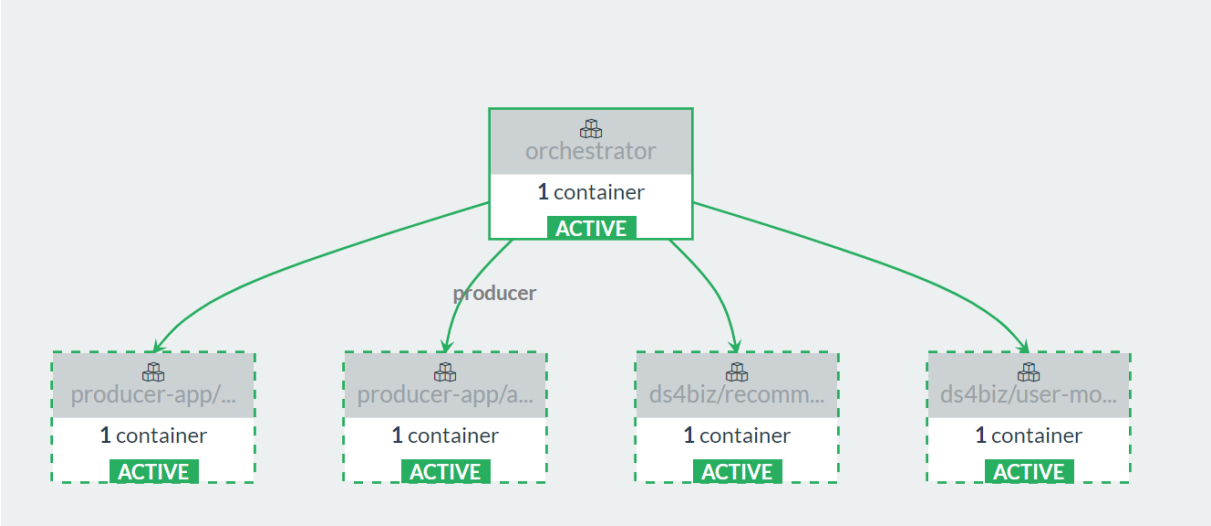


Figure 3: CPN Orchestrator integration

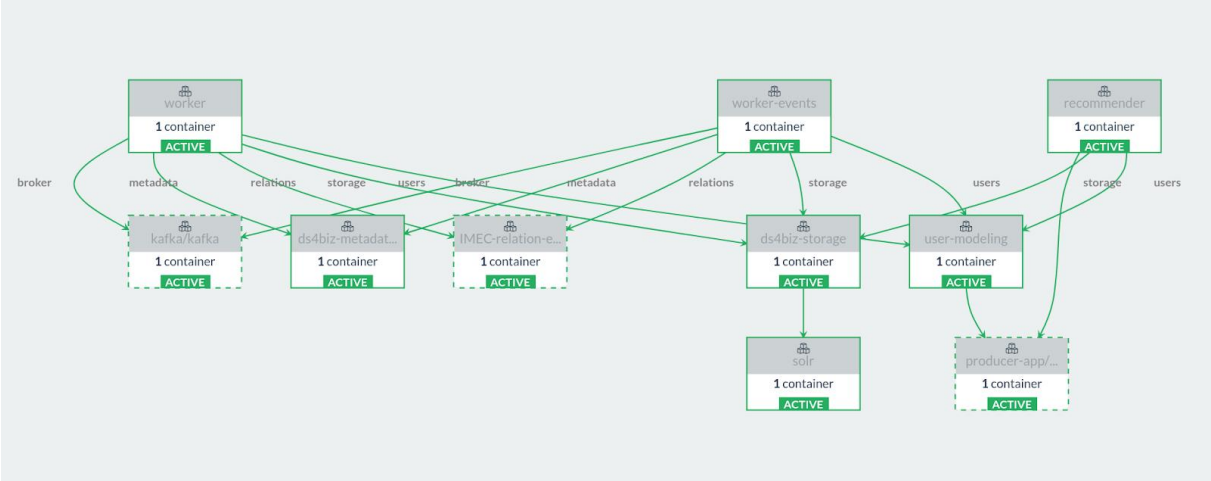


Figure 4: CPN Recommender Engine Integration

In the second prototype, the following APIs are "orchestrated" within the CPN platform:

- **/v1/add-action,** provides the user action data within the platform
- **/v1/recommend,** retrieves recommended contents
- **/v1/all,** retrieves the latest contents
- **/v1/popular,** retrieves the popular contents
- **/v1/update-timeframes,** updates the user timeframes

Co-funded by the Horizon 2020 Framework Programme of the European Union

2.2 CPN TECHNOLOGY BRICKS

In the second version of CPN platform, five new technology bricks were deployed and integrated and other ones were updated, as reported on D3.3 CPN Technology Bricks v2.

In particular, a new technology bricks not initially foreseen was integrated. In fact, the «Recommender AB-Testing» module, provided by LiveTech, has replaced the Sentiment Analysis module. This module was initially thought to analyze the "sentiment" associated to news articles as expressed by users on Social Networks (e.g. Facebook, LinkedIn, Twitter). Taking into account the users' requirements and the difficulty to extract data from social networks (due to GDPR regulation), the Consortium has decided to put more effort on the development of the Recommender AB-Testing.

In the following section the list of technology bricks that have been updated or delivered in the second prototype:

Layer	Name	Partner	Status
Content Technology Bricks	Semantic Lifting	Imec	New
	Topic Extractor	LiveTech	New
	Uplifting/Depressing Article Classifier	Imec	New
	Recommender AB-Testing	LiveTech	New
Mapping technology Bricks	Twitter Analytics - TRUTHNEST	ATC	New
	Producer's App	ENG	Updated
	Reader's App	ATC	Updated
	Recommender	LiveTech	Updated
	Personal Data Receipt	DCat	Updated

Table 2: CPN Technology Bricks for second prototype

Co-funded by the Horizon 2020 Framework Programme of the European Union

3 CPN PLATFORM – 2ND PROTOTYPE DELIVERY

The SCRUM agile process for delivering the platform, adopted for the first prototype, was also applied for the second one.

The process consisted of 3 phases: two starting activities steps and one cyclic step for task development. In detail:

- **Requirements prioritization**
- **Mapping Requirements/Technology Bricks**
- **Sprints (Implementation Results)**

3.1 REQUIREMENTS PRIORITISATION

The first step of the process was the requirements prioritization. In this phase starting from the list of requirements expected for the 2nd pilot iteration, as described in D1.4, the media partners (VRT, DW, DIAS), that in the case of the Scrum methodology [8]assumed the role of Product Owners, gave a priority for each user requirement expected. The result of this activity was an ordered list of requirements, named Prioritized Backlog for Pilot 2.

Below the Prioritized Backlog:

Priority	Requirement ID	Description
1	UR-UP2.5	The system should allow the user to down-/upload their network connections through user account.
2	UR-AF6.2	The system should allow users to consume content beyond their predefined timeframe after an interaction with the user (talkback)
3	UR- UP3.1	The system must allow the user to choose a preferred time frame or frames to consume content
4	UR- UP3.2	The system should create/refine time frames based on the user's consumption habits

[8] http://scrummethodology.com/

Co-funded by the Horizon 2020 Framework Programme of the European Union

5	UR- UP3.3	The system should refine the user's time frames through frequent interaction with the user (talkback)
6	UR- UP3.6	The system must allow the user to ignore a time frame completely
7	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 length and of what format it offers to the user
8	UR- UP1.4	The system should refine the user's interests through frequent interaction with the user (talkback)
9	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
10	UR-UP6.2	The system must keep track of how much of each item users consume, where they stop, continue and what they skip
11	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
12	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
13	UR-UP5.5	The system must give the user an easy option to agree to or withdraw from using location data for personalised offers
14	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
15	UR- UP5.2	The system should allow the user to set a home/main interest location
16	UR-UP5.4	The system should try to determine the surroundings of the user based on either

Co-funded by the Horizon 2020 Framework Programme of the European Union

		just location data or location data and direct interaction with the user (talkback)
17	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
18	UR-UP9.4	The user must be able to change and overwrite settings in their profile
19	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
20	UR-AF5.1	The system must offer the user an easy to access and easy to understand overview of their profile
21	UR-AF2.1	The system should show users who else from their network has consumed the same content item.
22	UR-AF2.2	The system should show users what else their network has shown, if there are differences
23	UR-AF2.3	The system should be able to show users the content item from another user (anonymously)
24	UR-AF5.2	The system must offer users easy access to their profile in order to change settings and data
25	UR-AF8.1	The system should allow users to search for specific topics they are temporarily interested in

Co-funded by the Horizon 2020 Framework Programme of the European Union

26	UR- AF5.3	The system must make it transparent to the users why they are shown certain content, based on an item level
27	UR-AF8.2	The system should allow users to add this search as a temporary personalisation category
28	UR-AF8.3	The system should allow users to define a specific time frame for this temporary change
29	UR-AF9.1	The system should allow users to define keywords and logical combinations of them to exclude content from their personalisation
30	UR-AF9.2	The system should allow users to define a time frame per keyword/logical combination
31	UR-AF9.3	The system should be able to overwrite this exclusion for important breaking rules
32	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
33	UR-AF6.3	The system should allow users to actively save articles for later consumption
34	UR-UP2.1	The system should allow for social media integration to recommend content based on what connections like, read and share UR-UP2.6 The system should allow users to search for other users on social media to build direct connections

Co-funded by the Horizon 2020 Framework Programme of the European Union

35	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)
36	UR-UP2.3	The system should allow for social media integration to keep track of what the user has already seen elsewhere.
37	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 on social media to prioritize the users for the personalisation
38	UR- UP2.5	The system should allow the user to down-/upload their network connections through user account.
39	UR- UP2.6	The system should allow users to search for other users on social media to build direct connections
40	UR-AF6.1	The system must allow users to access content again that they have already opened before
41	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)
42	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

Co-funded by the Horizon 2020 Framework Programme of the European Union

43	UR- AF7.2	The system should include guided feedback for specific elements of the system, allowing users to (help) improve it
44	UR-AF3.7	The system should be able to give the user a timeline overview of events regarding a specific topic
45	UR-PS1.1	The system should show the access to items through users by numbers (who, when, how long)
46	UR-PS1.3	The system should show which topics were most interesting to users
47	UR- PS2.4	The system should allow producers to export the record of their publications through standardized and interoperable formats
48	UR- PS2.5	The system should allow for an easy contribution of content from different publishers through standardised interfaces
49	UR- PS2.7	The system should allow editors to easily add missing attributes to articles manually
50	UR-UP3.7	The system should learn from these user responses and adjust its offerings accordingly

Co-funded by the Horizon 2020 Framework Programme of the European Union

Table 3: Prototype 2 prioritized requirements

3.2 MAPPING REQUIREMENTS/TECHNOLOGY BRICKS

The second step after the definition of Prioritized Backlog was mapping between these requirements and Technology Bricks. This phase involved both media and technical partners, which, in the Scrum Framework, represent the Scrum Team.

Starting from the technical requirements defined on D1.4, the Technology Bricks listed on section 2.2 and the core components of the platform addressed a list of technical tasks defined in this phase.

Below the list of task completed, linked with the respective user requirement, when expected:

Requriment ID	Task	Technology Brick
PS1.3	TPS1.3.1 - Producer's app must show to producers analytics on topics most interesting	Producer's App
PS2.4	TPS2.4.1 - Producer's app must provide an export functionality for the producer's contents	Producer's App
UP9.3	TUP9.3.2 - The PDR module should allow users to request the PDR again	Personal Data Receipt
General	T2.5 - Deploy component to extract additional data to enhance knowledge graph	Uplifting/Depressing Article Classifier
AF5.2	TAF5.2.1 - Reader's App - Edit Profile page.	Reader's App
UP9.5	TUP9.5.1 - Reader's App - Download profile data as JSON string	Reader's App
General	T2.4 - Create component to extract additional data to enhance knowledge graph	Uplifting/Depressing Article Classifier
General	Recommender have to push notification on client application during timeframes with a list of recommender articles	Recommender
General	Update the copy of the PDR	Personal Data Receipt
UP3.1	TUP3.1.2 - Recommender must allow the user to choose a preferred timeframe or frames to consume content, to postpone it and to ignore it.	Recommender

Co-funded by the Horizon 2020 Framework Programme of the European Union

General	Recommender must provide a list of articles that fit with the timeframes of the user	Recommender
General	API Gateway must expose Semantic Lifting APIs	API Gateway
General	API gateway must expose PDR APIs	API Gateway
General	New environment for second version of the CPN platform	CPN Platform
General	First deploy and integration of semantic lifting module	Semantic Lifting
General	Recommender must provide recommendation info (for transparency) with the list of recommendations	Recommender
General	T2.9 Add API to store and fetch rules for knowledge graph extraction	Uplifting/Depressing Article Classifier
General	APIs for popular and recent stream have to provide "read" field in output	API Gateway
PS1.1	TPS1.1.2 - Producer's app must show to producers analytics on items consumed	Producer's App
PS1.1	TPS1.1.1 - Producer's app must provide an UI for producer's	Producer's App
General	API Gateway must expose APIs that requires only the bearer token and not the userId	API Gateway
General	API Gateway must expose services for recommender management	API Gateway – Recommender AB-Testing
General	CPN Admin API: Recommender Management (CRUD operations) API	Recommender AB-Testing
General	Producer's App Must expose a service to search all articles (filtered by media provider	Producer's app
AF2.5	TAF2.5.1 - Recommender must offer more content once all proposed articles have been consumed.	Recommender
AF8.1	AF8.1.1 - Reader's App - Topics search user interface	Reader's App
AF8.2	AF8.2.1 - Reader's App - Personalization categories	Reader's App

Co-funded by the Horizon 2020 Framework Programme of the European Union

AF8.3	AF8.3.1 - Reader's App - Time frame for personalization category	Reader's App
AF9.1	AF9.1.1 - Reader's App - Support queries with keywords	Reader's App
AF9.2	AF9.2.1 - Reader's App - Define time frames per keyword	Reader's App
AF7.2	AF7.2.1 - Reader's App - Offer guided feedback to specific elements	Reader's App
General	Redefine user events types and attributes	API Gateway – Reader's App – Recommender – Orchestrator – Message Broker
General	Reader's app must show recommender reasons in articles card	Reader's App
UP2.1	UP2.1.1 - Reader's App - Create social media profile	Reader's App – Twitter Analytics
UP2.2	UP2.2.1 - Reader's App - Social media profile should contain liked/ shared articles information	Reader's App – Twitter Analytics
AF2.1	AF2.1.1 - Reader's App - Show users from their network that have consumed the same content.	Reader's App – Twitter Analytics
AF2.2	AF2.2.1 - Reader's App - Show what else their network has shown, if there are differences	Reader's App – Twitter Analytics
UP2.3	UP2.3.1 - Reader's App - Social Media profile should contain information of user's network news preferences.	Reader's App – Twitter Analytics
UP2.4	UP2.4.1 - Reader's App - Social media profile should contain user network interactions details	Reader's App – Twitter Analytics
General	Reader's App must allow like/dislike with "swipe" action	Reader's App
UP6.3	TUP6.3.2 - API Gateway must expose a service to handle user feedback	API Gateway
General	T2.1 - Deploy knowledge graph extraction brick on CPN platform	Uplifting/Depressing Article Classifier
General	Reader's App: On register screen, modify allow permission to "time".	Reader's App

Co-funded by the Horizon 2020 Framework Programme of the European Union

UP9.2	UR-UP9.2 Reader's App: show text to require users consent explicitly.	Reader's App
General	Notification message triggered by the Recommender.	Message Broker – Recommender
AF6.3	AF6.3.1 - Reader's App - Save articles	Reader's App
General	T2.6 Try semantic uplifting on sample of Live Tech	Uplifting/Depressing Article Classifier - Recommender
General	T2.7 Add documentation to knowledge graph extraction brick	Uplifting/Depressing Article Classifier
UP6.2	UP6.2.1 - Reader's App - Track read time and scrolling depth of an article.	Reader's App
UP6.2	TUP6.2.2 - API Gateway must expose a service to register read time and scrolling depth	API Gateway
UP6.3	UP6.3.1 - Reader's App - Trigger talkback message on skipping content	Reader's App
AF6.2	UR- AF6.2.1 - Reader's App - The system should allow users to consume content beyond their predefined time- frame after an interaction with the user (talkback)	Reader's App
UP3.3	TUP3.3.1 - Reader's app should support refining the user's time frames through frequent interaction with the user (talkback)	Reader's App
UP3.3	TUP3.3.2 - Reader's App - Support interaction to refine the user's time frames (talkback).	Reader's App
UP5.5	UP5.5.1 - Reader's App - Display permission for using location data	Reader's App
General	Reader's App: Improve loading speed.	Reader's App
UP3.2	TUP3.2.1 - Reader's app must track timeframes of user activity habits.	Reader's App
UP3.1	TUP3.1.1 - Reader's app must allow the user to choose a preferred time frame or frames to consume content, to postpone it and to ignore it.	Reader's App

Co-funded by the Horizon 2020 Framework Programme of the European Union

UP3.1	TUP3.1.3 - API Gateway - Must expose a service to update the user timeframes	API Gateway
UP9.4	UP9.4.2 - Reader's App - Edit user profile settings	Reader's App
UP9.3	UP9.3.1 - Reader's App - Edit user usage related data.	Reader's App
General	T2.2 - Integrate knowledge graph extraction brick with other components	Semantic Lifting – Recommender
General	Integration of semantic uplifting component in recommender cycle	Uplifting/Depressing Article Classifier

Table 4: 2nd prototype technical tasks

3.3 RESULTS

The third step was the implementation of the tasks in an iterative way. Following the Scrum Framework, a series of two-weeks sprints was defined and at the start/end of each sprint a technical conference call was arrange in order to define the next sprint plan, the next sprint execution and the previous sprint review.

As results of these activities, the second platform was delivered including almost the whole of the requirements expected for the second pilot iteration. The requirements not included in this iteration will be moved on the third version of the platform.

The following paragraphs show the list of requirements completed the status of the platform and the list of functionalities provided.

3.3.1 Completed requirements

Requirement ID	Description	Status
UR-UP2.1	The system should allow for social media integration to recommend content based on what connections like, read and share	Completed
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)	Completed
UR-UP2.3	The system should allow for social media integration to keep track of what the user has already seen elsewhere	Completed

Co-funded by the Horizon 2020 Framework Programme of the European Union

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 on social media to prioritize the users for the personalisation	**Completed**
UR-UP2.5	The system should allow the user to down-/upload their network connections through user account.	**Completed**
UR- UP3.1	The system must allow the user to choose a preferred time frame or frames to consume content	**Completed**
UR- UP3.2	The system should create/refine time frames based on the user's consumption habits	**Completed**
UR- UP3.3	The system should refine the user's time frames through frequent interaction with the user (talkback)	**Moved to 3rd Prototype**
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 length and of what format it offers to the user	**Completed**
UR- UP3.7	The system should learn from these user responses and adjust its offerings accordingly	**Moved to 3rd Prototype**
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	**Completed**
UR- UP5.2	The system should allow the user to set a home/main interest location	**Partially**
UR-UP5.5	The system must give the user an easy option to agree to or withdraw from using location data for personalised offers	**Completed**
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	**Completed**

Co-funded by the Horizon 2020 Framework Programme of the European Union

UR-UP6.2	The system must keep track of how much of each item users consume, where they stop, continue and what they skip	**Completed**
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	**Completed**
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	**Completed**
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	**Completed**
UR-UP9.4	The user must be able to change and overwrite settings in their profile	**Completed**
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	**Completed**
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	**Completed**
UR- AF2.5	Once all articles proposed have been consumed, the system should only offer more content upon request by the users	**Completed**
UR-AF5.1	The system must offer the user an easy to access and easy to understand overview of their profile	**Completed**
UR-AF5.2	The system must offer users easy access to their profile in order to change settings and data	**Completed**
UR- AF5.3	The system must make it transparent to the users why they are shown certain content, based on an item level	**Completed**

Co-funded by the Horizon 2020 Framework Programme of the European Union

UR- AF6.2	The system should allow users to consume content beyond their predefined time- frame after an interaction with the user (talkback)	Moved to 3rd Prototype
UR-AF6.3	The system should allow users to actively save articles for later consumption	Completed
UR- AF7.2	The system should include guided feedback for specific elements of the system, allowing users to (help) improve it	Completed
UR-AF8.1	The system should allow users to search for specific topics they are temporarily interested in	Completed
UR-AF8.2	The system should allow users to add this search as a temporary personalisation category	Completed
UR-AF8.3	The system should allow users to define a specific time frame for this temporary change	Completed
UR-AF9.1	The system should allow users to define keywords and logical combinations of them to exclude content from their personalisation	Completed
UR-AF9.2	The system should allow users to define a time frame per keyword/logical combination	Completed
UR-AF9.3	The system should be able to overwrite this exclusion for important breaking rules	Completed
UR-PS1.1	The system should show the access to items through users by numbers (who, when, how long)	Completed
UR-PS1.3	The system should show which topics were most interesting to users	Completed
UR- PS2.4	The system should allow producers to export the record of their publications through standardized and interoperable formats	Completed

Co-funded by the Horizon 2020 Framework Programme of the European Union

UR- PS2.5	The system should allow for an easy contribution of content from different publishers through standardised interfaces	**Moved to 3rd Prototype**
UR- PS2.7	The system should allow editors to easily add missing attributes to articles manually	**Moved to 3rd Prototype**

Table 5: 2nd prototype completed requirements

3.3.2 Status of the platform

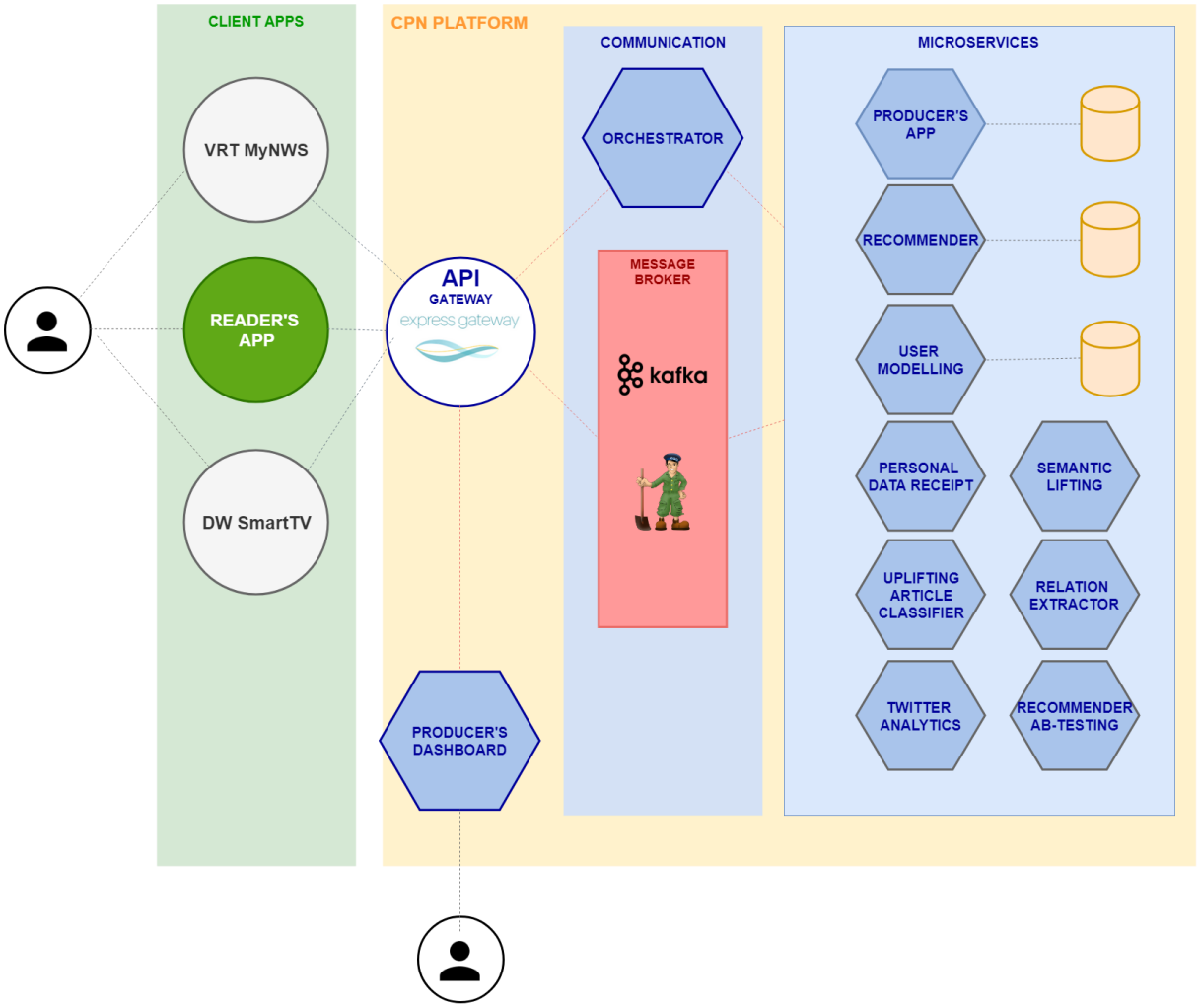
The second CPN platform prototype was deployed and released in a new dedicated environment, ready to be exploited for the second pilot phase. The oldest environment will be used for development and testing.

This second version includes:

- **10 Technology Bricks deployed as microservices (9 backend services + 1 web dashboard)**
- **1 Technology Brick as client application, in web and mobile version (Reader's App)**
- **3 Core components**

Co-funded by the Horizon 2020 Framework Programme of the European Union

The result of the second prototype released is represented in the figure below:



More in detail, the following macro-functionalities were implemented in this second prototype:

- **Definition and update of timeframes**
- **Configuration of the recommender and definition of A/B testing environments**
- **Update of the User profile and of the Personal Data Receipt**
- **Collection and aggregation of user actions for analytics and insights**
- **Uplifting and semantic analysis of the contents**
- **Integration of Twitter user profiles**
- **Providing information to users on recommendations for greater transparency**

Co-funded by the Horizon 2020 Framework Programme of the European Union

4 PLATFORM INTEGRATION – SIDETRACKS

In addition to the development and deployment activities, a first phase of integration of other client applications within the CPN platform was conducted.

In particular, two internal activities, in collaboration with two partners of the consortium (VRT and DW), led to the integration of two other client applications: a new mobile website and a smart Tv app.

In parallel, other media companies interested in the functionality offered by the CPN platform were contacted and a first technical integration phase was started with three of them: Agence France Presse, Groupe Sud Ouest and RCS Media Group.

In order to perform these integration activities, a document with the guidelines for the media companies' integration was elaborated and is reported as Annex A of this deliverable.

4.1 VRT SIDETRACK – VRT MYNEWS

VRT performed a four-week sidetrack experiment with over 900 registered users. The users tried an in-house developed mobile website, which offered personalized news from VRTNWS. The backend used several CPN microservices, amongst which the recommender system and the registration module. The mobile website was similar to VRTNWS, but had some simplifications.

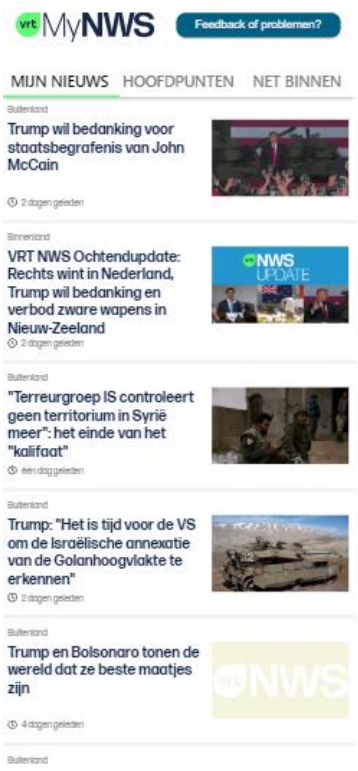


Figure 5: VRT MyNws UI

Co-funded by the Horizon 2020 Framework Programme of the European Union

During the full four-week period, there was a strong decrease in active users. By the end of the trail, less than 20% of the users visited the website every day. Users could be reactivated through emails, which we did three times. The effect lasted for only one or two days, however. We believe this is normal behaviour, which should be taken into account for when designing future experiments.

During the first phase, we performed an A/B-test with two recommender configurations. The recommender system was a hybrid of collaborative filtering, content-based recommendations, and randomly selected news items. The varied parameter was the mixing ratio between the three underlying system.

During the second phase, we shifted our focus to finding the best baseline system. The baseline system chosen is the popularity recommender. This system gives the most read articles as a list to each user. Currently, VRTNWS.be (the main website, not this experiment) has a stream "MEEST GELEZEN" (MOST READ) that contains the 20 articles that were most read during the last 12 hours. This is an arbitrary choice, and we wondered if it was optimal. We divided the users into three groups, that received lists of articles that were most read during the last hour, the last twelve hours and the last two days, respectively. The first group clearly was more satisfied: they had more overall clicks (in any stream), and a higher fraction of clicks in the recommended stream.

We equally performed qualitative research through surveys. At the time of writing, the survey results are being analysed.

4.2 DW SIDETRACK – DW SMARTTV APP

The DW Sidetrack was the first experimentation of the platform with another type of media, namely video recommendation. DW had already developed a smart TV prototype app that was including a very simple recommendation engine based on keywords inserted in the video description. The user profile was stored directly on the customer device and therefore there was no global visibility of users' interest for the media company.

They are currently working on the integration of the CPN recommender services in this app.

The process of integration has been outlined as follows:

1. Users perform their regular login on the SmartTV app
2. Upon login an app callback tries to login into the CPN app or automatically create a new user account (if such account doesn't already exist)
3. In the "Recommended videos" section the app is showing the result of the call to CPN recommended services

In order to configure this integration the following actions have been undertaken on CPN back-end side

Co-funded by the Horizon 2020 Framework Programme of the European Union

1. Configuration of a new ingestion task to retrieve videos from DW streams

2. Configuration of a recommender for “DW-Video” media account

The recommender chosen for this kind of media has been a hybridization of a content-based recommender based on video annotation keywords and a collaborative user-item recommender. The former approach, although the annotation keywords are fairly simple and generic, is suitable especially in overcoming the cold start problem. As soon as a user starts consuming videos, the keywords are collected and stored in the centralized CPN user-profiling module and are exploited for filtering next videos to be proposed to the user. On the other hand, the collaborative filtering approach is getting better overall results given that the user base is large enough and has a longer history of videos consumption. It must be noted that collaborative filtering approaches are “media content” independent so they are able to capture more subtle user preferences (by comparing similar patterns of consumption among user).

The integration is currently an ongoing process that is performed at “best effort” inside DW Smart Video department and it is planned to be completed in 1-2 business months.

As a final note, it is worth to mention that other content-based approaches are being evaluated:

1. Analysis of video content by using text-to-speech techniques in order to be to analyse the textual content of the video

2. Video analysis by using the latest advancements, especially in Deep Learning techniques (Object Detection and Tracking models, Video Tagging and Topic annotation) to extract video annotations directly from the sequence of images

These approaches are currently beyond the scope of CPN project but their feasibility is being currently being carefully considered since they could enable significant business opportunities. Possible approaches to the realization of such features include in-house development, outsourcing to external companies (e.g. Open Innovation, Hackathons) and raising additional funding from external investors or other public funding instruments.

Co-funded by the Horizon 2020 Framework Programme of the European Union

5 CONCLUSIONS

This document represents the report of all the activities conducted for the release of the second version of CPN Open Virtual platform.

The CPN platform v2 was packaged and it is available for the partners in internal repository.

In order to test the platform and verify the status of delivery as described in this report, the following software prototypes are available to internal partnership:

- **CPN Microservices Platform v2 – Two separated environments (one for Pilot 2 and one for development and testing)**
- **CPN API Gateway v2 - Two separated environments (one for Pilot 2 and one for development and testing)**
- **Producer's Dashboard v1**
- **Android Reader's App v1**
- **Web Reader's App v2**

The next activities will be focused on Technology Bricks and platform improvements, in order to satisfy the complete set of the requirements and release the complete CPN platform at the end of the year (CPN Open Virtual platform v3 is expected on 31st December 2019)

Co-funded by the Horizon 2020 Framework Programme of the European Union

6 REFERENCES

[1] https://www.projectcpn.eu/s/CPN_D22_CPN_Open_Virtual_Platform_v1_20180629_v10.pdf

[2] https://www.projectcpn.eu/s/D33-Technology-Bricks-V2.pdf

[3] https://www.projectcpn.eu/s/CPN_D14_Technical_requirements_platform_and_service_requirements_20180830_v10.pdf

[4] https://expressjs.com/

[5] https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md

[6] https://swagger.io/

[7] http://scrummethodology.com/

Co-funded by the Horizon 2020 Framework Programme of the European Union

7 ANNEX A

HowTo - Media Company Integration

1

Co-funded by the Horizon 2020 Framework Programme of the European Union

1. Step 1 - Articles Injection

The CPN platform offers personalization and recommendation functionalities in as-a-service mode. In order to do this, it needs to analyze the content coming from media sources that want to access these functionalities.
The first step to follow for the integration of a media company within the platform is to provide access to its contents.

The CPN platform and in particular the Technology Brick Producer's App, takes care of integrating the contents within the platform, normalizing them and making them available to all the other components through the Apache Message broker. The media company can provide its own contents in two different ways:

1. **RSS feed**
2. **REST service**

At the moment the CPN platform provides, in addition to the recommended contents, also the most recent and the most popular contents. If you want to provide your users with these types of content, at least two data sources are required:

- **A list with all the articles** (Always mandatory)
- **A list with the most popular articles** (Optional if you want to provide more content lists)

The following input fields are mandatory or strongly recommended:

```
originId : String, // A unique ID for identify the article Mandatory
url : String //The article URL Mandatory
title : String,    Mandatory
text : String,   // Summary  Strongly recommended
content: String, // Full text article    Strongly recommended
date : Date, // date in Date format Mandatory
tags : [String] //list of tags, Strongly recommended
category : String, // Category or Topic of the article Strongly recommended
```

The ouput format of normalized articles will be the following:

Co-funded by the Horizon 2020 Framework Programme of the European Union

2. Step 2 - Client-side Integration

The second step is focused on the integration of the CPN platform services within the client-side application of the media company.
The integration for the client applications is totally API based. This allows the integration of recommendation services in any type of client application. At the moment the CPN platform has been successfully tested with web, mobile and smart Tv apps, but can support any type of them.

This second step is divided into three parts:

- User identification
- Integration of recommendation services
- Integration of user profiling services

1.2 User Identification

A mandatory step of this process is the identification of the user within the CPN platform. In fact, all the personalisation services offered by CPN are strictly related to a specific user.

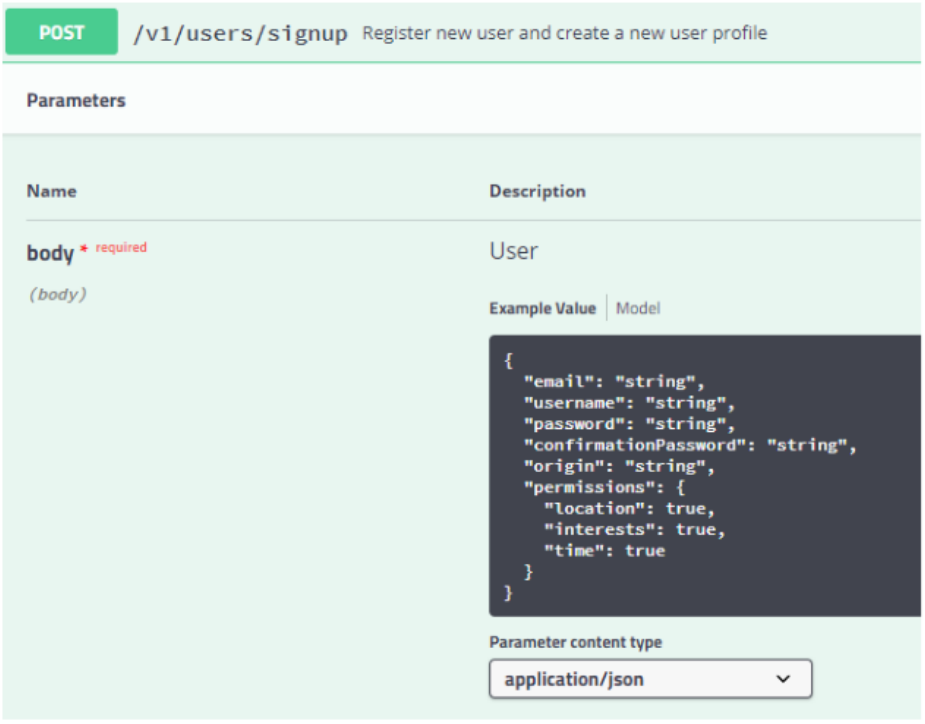
At the moment, the only way to identify a user is through a registration system but since the CPN platform do not use any personal data of the users, will be implemented soon altri other identification systems (e.g. cookie-based or some fingerprint).

The CPN platform provides two different APIs for Signup and Login:

Signup

It registers a specific user to the CPN platform. The specification are shown in the figure below:

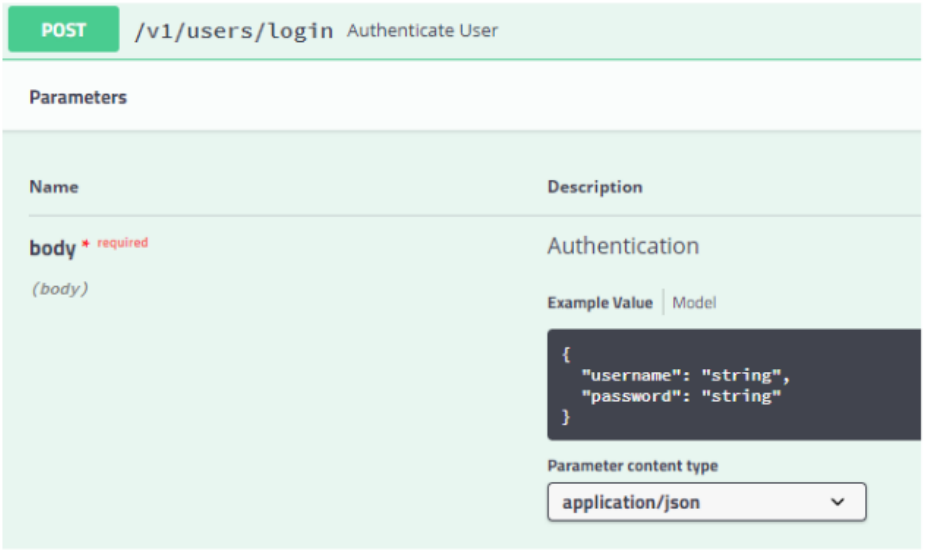
Co-funded by the Horizon 2020 Framework Programme of the European Union



Login

It identifies a specific user and returns a **JSON Web Token**. The **token is required** to access all the personalised services. The specifications are shown in the figure below:

Co-funded by the Horizon 2020 Framework Programme of the European Union



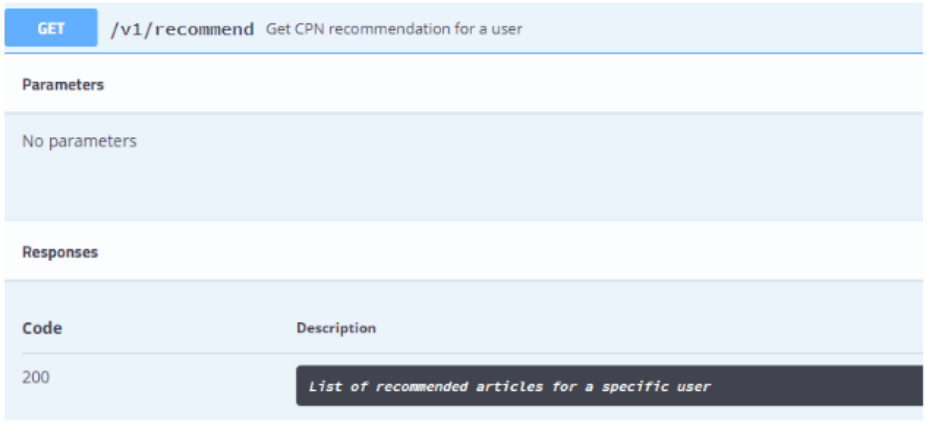
All the CPN APIs (apart the signup and login ones) needs a JWT for user identification. In order to access a protected API, the user agent should send the JWT in the Authorization header using the Bearer schema. The content of the header should look like the following:

```
Authorization: Bearer <token>
```

1.3 RECOMMENDATIONS

The CPN platform provide a protected API for recommendation service. This API returns a list of articles recommended for a specific user.

Co-funded by the Horizon 2020 Framework Programme of the European Union



The API **doesn't need input parameter** but as mentioned before the **JWT is mandatory** in order to identify the user.

Below an example of a recommended item:

```
recommendedArticle:
{
        _id : String, // Internal Id
        originId : String, // Original Id from media source
        origin : String, // Source
        url : String //original item url
        completeUrl : String //complete original item url (in case of minified URL)
        category : String, // Category of article
        multimedia : [{ // list of media objects (images, video, etc..)
                            type: String,
                            url: String,
                            name: String,
        }]
        title : String,
        text : String,    // Summary
        content: String, // Full text article
        language : String,
        author : String,
        date : Date, // date in Date format
        dateStr : String,    // date in String format
```


Co-funded by the Horizon 2020 Framework Programme of the European Union

```
        timestamp : Number, // date in timestamp format
        location : { //location in geoJson format
                type: Object,
                index: '2dsphere',
                sparse: true },
        tags : [String] //list of tags,
        popular: Boolean, // If this article become from popular feed,
        read: Boolean, // If the user already read the article
        score: Number, // The score of the recommendation of the article
        recommender_id: String, //The id of the recommender that produces this recommendation
        Description: String //Human readable description of the recommendation (for
        transparency)
}
```

1.4 User Profiling

In order to provide more precise personalized content for a specific user, the CPN platform needs to create a profile for each user and enrich these profiles while them using the CPN platform.

The users' behaviour is tracked by the CPN platform and now many "user actions" are collected:

1. "Interested"
2. "Not interested"
3. "Read"
4. "Rate"
5. "Share"
6. "Interest feedback"

1. "Interested" action

```
{
 "event": "Favorite item from stream: PERSONALIZED",
 "userId": "5b8fe613e7cffb000a11bafa",
 "itemId": "5c545800c13b2f1e00310b91",
 "timestamp": "1549443058160",
```


Co-funded by the Horizon 2020 Framework Programme of the European Union

```
  "info": {
    "latitude": "38.0347891",
    "longitude": "23.7930435",
    "accuracy": "951"
    }
}
```


The /v1/add-action method will parse the event string and extract the action and stream. We could split the string to two fields (e.g. event: “Favorite item” and stream: “PERSONALIZED”) but currently this is how it works.

The location information inside the info section appear because this user enables the geolocation. Otherwise, the response would be empty:


```
{
  "event": "Favorite item from stream: PERSONALIZED",
  "userId": "5b8fe613e7cffb000a11bafa",
  "itemId": "5c545800c13b2f1e00310b91",
  "timestamp": "1549443058160",
  "info": { }
}
```


2. "Not interested"


```
{
  "event": "Remove item from stream: PERSONALIZED",
  "userId": "5b8fe613e7cffb000a11bafa",
  "itemId": "5c57439cc13b2f1e0032d88c",
  "timestamp": "1549444489459",
  "info": {
    "latitude": "38.0347891",
    "longitude": "23.7930435",
    "accuracy": "951"
  }
}
```


3. "Read"


```
{
```


Co-funded by the Horizon 2020 Framework Programme of the European Union

```
 "event": "Read item from stream: PERSONALIZED",
 "userId": "5b8fe613e7cffb000a11bafa",
 "itemId": "5c574146c13b2f1e0032d712",
 "timestamp": "1549444639401",
 "info": {
  "readTimeMillis": "6235",
  "fullRead": true
 }
}
```


The info section contains the fields readTimeMillis, which corresponds to the (total) read time of this article in milliseconds and fullRead (boolean), if the user has scrolled to the bottom of the article.

4. "Rate"


```
{
 "event": "Favorite item from stream: PERSONALIZED",
 "userId": "5b8fe613e7cffb000a11bafa",
 "itemId": "5c55a26fc13b2f1e0031cf71",
 "timestamp": "1549448219381",
 "info": {
  "rating": 1,
  "ratingComment": ""
 }
}
```


The rating is an integer from 1 (Not at all relevant) to 5 (Very relevant) and rating comment is a string field (containing the user comments), as requested by VRT.

5. "Share"


```
{
 "event": "Share item to Twitter from stream: PERSONALIZED",
 "userId": "5b8fe613e7cffb000a11bafa",
 "itemId": "5c51ca0ec13b2f1e002f7443",
 "timestamp": "1549448912325",
 "info": {
  "latitude": "38.0347891",
```


Co-funded by the Horizon 2020 Framework Programme of the European Union

```
  "longitude": "23.7930435",
  "accuracy": "951"
 }
}
```


The event can also be "Share item to Facebook", "Share item to LinkedIn", "Share item to Google+".

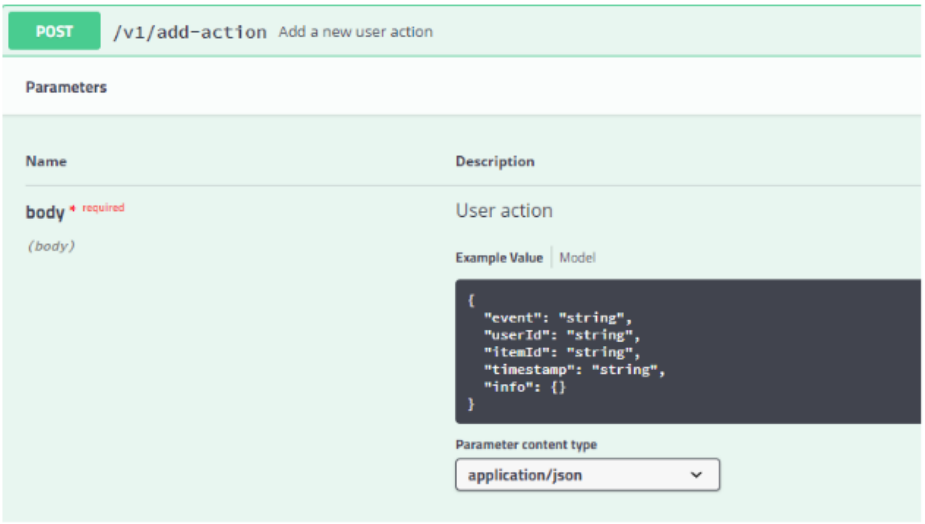
6. "Interest feedback"


```
{
 "event": "Interest feedback",
 "userId": "5b7ea4fce7cffb000a11baf8",
 "itemId": "5c3f0c64c13b2f1e00240fa5",
 "timestamp": "1549451022047",
 "info": {
  "interestFeedback": "No more football news!"
 }
}
```


The interestFeedback text contains the user-entered text. This type of action (feedback) is subject to change.

The specification are shown in the figure below:

Co-funded by the Horizon 2020 Framework Programme of the European Union



Co-funded by the Horizon 2020 Framework Programme of the European Union