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

Deliverable 1.2
Standard Tool for Information Gathering

Work Package 1: Data Protection, Ethical Impact and Interoperability

affecTive basEd iNtegrateD carE for betteR Quality of Life: TeNDER Project

Grant Agreement ID: 875325

Start date: 1 November 2019

End date: 31 October 2022

Funded under programme(s): H2020-SC1-DTH-2018-2020/H2020-SC1-DTH-2019

Topic: SC1-DTH-11-2019 Large Scale pilots of personalised & outcome based integrated care

Funding Scheme: IA - Innovation action

Disclaimer

The TeNDER consortium consists of the following Partners.

Table 1 - Consortium Partners List

No	Name	Short name	Country
1	UNIVERSIDAD POLITECNICA DE MADRID	UPM	Spain
2	MAGGIOLI SPA	MAG	Italy
3	DATAWIZARD SRL	DW	Italy
4	UBIWHERE LDA	UBI	Portugal
5	ELGOLINE DOO	ELGO	Slovenia
6	ETHNIKO KENTRO EREVNAS KAI TECHNOLOGIKIS ANAPTYXIS	CERTH	Greece
7	VRIJE UNIVERSITEIT BRUSSEL	VUB	Belgium
8	FEDERATION EUROPEENNE DES HOPITAUX ET DES SOINS DE SANTE	HOPE	Belgium
9	SERVICIO MADRILENO DE SALUD	SERMAS	Spain
10	SCHON KLINIK BAD AIBLING SE & CO KG	SKBA	Germany
11	UNIVERSITA DEGLI STUDI DI ROMA TOR VERGATA	UNITOV	Italy
12	SLOVENSKO ZDRUZENJE ZA POMOC PRI DEMENCI - SPOMINCICA ALZHEIMER SLOVENIJA	SPO	Slovenia
13	ASOCIACION PARKINSON MADRID	APM	Spain

Document Information

Project short name and Grant Agreement ID	TeNDER (875325)
Work package	WP 1
Deliverable number	D1.2
Deliverable title	Standard Tool for Information Gathering
Responsible beneficiary	UBI
Involved beneficiaries	VUB, SPO, UPM
Type[1]	Report
Dissemination level[2]	Public
Contractual date of delivery	30 June 2020
Last update	30 June 2020

[1] **R:** Document, report; **DEM:** Demonstrator, pilot, prototype; **DEC:** Websites, patent fillings, videos, etc.; **OTHER**; ETHICS: Ethics requirement; ORDP: Open Research Data Pilot.

[2] **PU:** Public; **CO:** Confidential, only for members of the consortium (including the Commission Services).

Document History

Version	Date	Status	Authors, Reviewers	Description
v 0.1	01/06/2020	Draft	Ricardo Vitorino (UBI)	Table of Contents
v 0.2	02/06/2020	Draft	Ricardo Vitorino, Ricardo Carrola (UBI)	Survey for data assessment
v 0.3	23/06/2020	Draft	Ricardo Vitorino, Rui Alheiro, Ricardo Carrola (UBI)	Standards in eHealth
v 0.4	28/06/2020	Draft	Ricardo Vitorino, Ricardo Carrola, Bruno Silva (UBI)	Architecture and Components description
V0.5	29/06/2020	Draft	Ricardo Carrola, Ricardo Vitorino (UBI)	Conclusions
V0.6	29/06/2020	Draft	Špela Glišović Krivec, David Krivec (SPO)	First Review
V0.7	30/06/2020	Draft	Lisa Feirabend, Paul Quinn (VUB)	Second review
V0.8	30/06/2020	Draft	Annelore Hermann (UPM)	General review
V0.9	30/06/2020	Draft	Ricardo Vitorino (UBI)	Consolidation of feedback from reviews and Executive Summary
V1.0	30/06/2020	Draft	Annelore Hermann (UPM)	General Review
V1.1	30/06/2020	Final	Gustavo Hernández	General Review for submission

Acronyms and Abbreviations

Acronym/Abbreviation	Description
API	Application Programming Interface
CEF	Connecting Europe Facility Programme
CRUD	Create, Read, Update and Delete operations
CSV	Comma-Separated Values
DICOM	Digital Imaging and Communications in Medicine
DSI	Digital Service Infrastructure
EC	European Commission
eEIF	eHealth European Interoperability Framework
eHDSI	eHealth Digital Service Infrastructure
EHR	Electronic Health Record
EIF	European Interoperability Framework
EU	European Union
FHIR	Fast Health Interop Resources
GUI	Graphical User Interface
HL7	Health Level Seven International
HLS	High-Level Subsystems
IHE	Integrating the Healthcare Enterprise
IoT	Internet of Things
MVC	Model-View-Controller
NGSI	Next-Generation Specification Interface
PDF	Portable Document Format
RDF	Resource Description Framework
ReEIF	Refined eHealth European Interoperability Framework
REST	Representational state transfer
SDK	Software Development Kit
XML	Extensible Markup Language

Contents

List of Figures

List of Tables

Executive Summary

This deliverable presents the methodology initially defined for the information gathering, from different sources and with different characteristics. In order to better define this methodology, a survey was created whose objective was to identify what data exist or will exist, for the different partners and solutions (existing and future ones), as well as the respective representation formats of that data and how these can be shared between the different modules and solutions.

At the same time, the standards for the data exchange in the context of eHealth were analysed at the state-of-the-art level, particularly the EC recommendations for the adoption of certain standards and reference architectures.

These two components (data and standards) made it possible to describe the first approach for the tool's architecture definition and its implementation, described in this document.

It should be noted that the architecture proposed in this document is an initial version and subject to changes, taking into account the continuous identification of requirements (also as a result of other tasks and work packages in progress in parallel), with the final version being consolidated in the deliverable D2.3.

1 INTRODUCTION

1.1 Purpose and scope

The TeNDER project aims at performing five large-scale pilots, involving different user partners and thousands of final users in four different European regions, which involve patients, health professionals (physicians mainly), social workers, caregivers and other staff (clerks, hospital IT support, etc.), to provide tailor suited integrated care services to promote wellbeing and health recovery. The main goal of these pilots, demonstrated in four different scenarios (home, day-care centres, rehabilitation centres/rooms, and hospitals), is to create an integrated care ecosystem for assisting people with chronic diseases of Alzheimer's Disease (AD), Parkinson's Disease (PD), Cardiovascular Diseases (CVD) and co-morbidity through the use of affect based micro-services and tools. These microservices will recognise the mood of a person and match with clinical data (from Electronic Health Records or EHRs) and clerical patient information, while preserving privacy, monitoring the ethical principles, providing data protection and security, with the result of an increased Quality of Life (QoL).

The present deliverable describes an integrated tool for gathering of background, medical and personal information, based on the state-of-the art requirements and committed in the integrated healthcare paradigm. Given that the collection of quantitative and qualitative data will take place in each of the countries involved in the pilots, with multiple aims, templates and methodologies, this tool needs to be integrated and usable in multiple ways related to services provided by TeNDER: health-care tracking, monitoring, interaction, quality of life assessment, among others. Being part of Work Package 1, the task outcomes are also aligned with European Regulations in terms of data privacy, security, integrity, and interoperability, a set of strong requirements that will support the design of TeNDER ecosystem.

1.2 Contribution to other deliverables

The present deliverable will contribute to newer versions of D1.1, as well as to updates performed to the consent forms and procedures made available in WP10 (currently D10.2, D10.3 and D10.8).

This proposed tool will be a baseline in D2.3 for the TeNDER architecture and is tightly linked to D5.3 (European Interoperable Health Records and Pathway Gathering).

1.3 Structure of the document

The document starts by describing the data collection needs from the pilot and the project side, then analysing the existing standards in the European ecosystem of information sharing within eHealth and finally towards the proposed software architecture of information gathering for TeNDER, explaining the need and usage for all the modules in it.

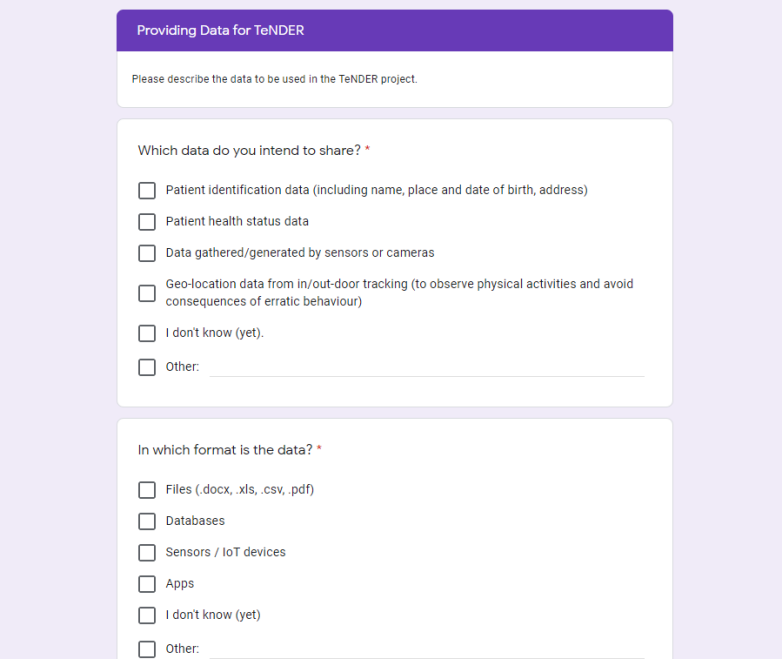
2 INFORMATION GATHERING

Considering that the collection of quantitative and qualitative data takes place in each of the countries involved in the pilots, with multiple aims, tools templates and methodologies, and with the purpose of being integrated and usable in multiple ways, the team started by requesting all the project partners to fill in a survey to assess which partners intend to share personal or clinical data with the project and to gather more details about the information (types, formats, ways of sharing).

At the moment of writing this deliverable, the technical architecture for the TeNDER platform was still under discussion and some components and information flows were still to be determined. Nonetheless, there was already an obvious link between this task and Task 5.1, responsible for the implementation the European regulation for data exchanging, based on existing Electronic Health Records (EHR) systems, a database to store the medical profile will extend information available from physical, medical and behavioural activity, which allows the information securely flowing from patients, to the system and to the health professionals.

2.1 Information Gathering Survey

Ubiwhere created a web form using Google Drive[3] and engaged all the TeNDER partners to assess if they would be willing or intending to share personal or clinical information with the project, and if so, to determine which types of data (as defined in deliverable D1.1: patient identification data, patient health status data, data gathered/generated by sensors or cameras, geo-location data from in/out-door tracking, others), the formats of the data (files, databases, sensors, apps, others) as well as the preferred ways of sharing such data (web interface, REST API, SDK, others), as shown in the image below.



[3] TeNDER Data Collection Google Form – https://docs.google.com/forms/d/e/1FAIpQLSed91hwIMAC7du5sjgrrlmuxlKL2ovm6G9UHE92rrYF-NmkjA/viewform?usp=sf_link

Figure 1 – Data Collection Google Form web interface

The survey collected 12 responses, from 11 partners, where more than half demonstrated positive willingness to share data with the project (cf. image below).

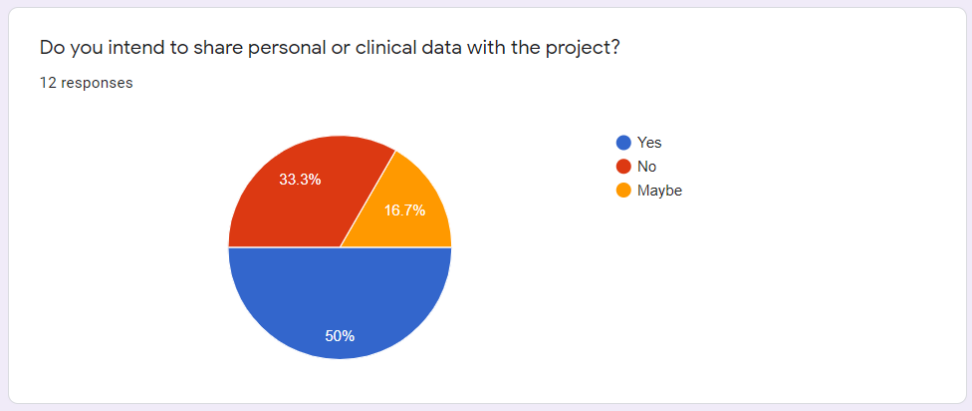


Figure 2 – Willingness to share personal or clinical data with the project

Considering such a positive assessment for data sharing with the project, the partners then declared which types of data they intend to share, as well as the formats these are currently stored (cf. image below).

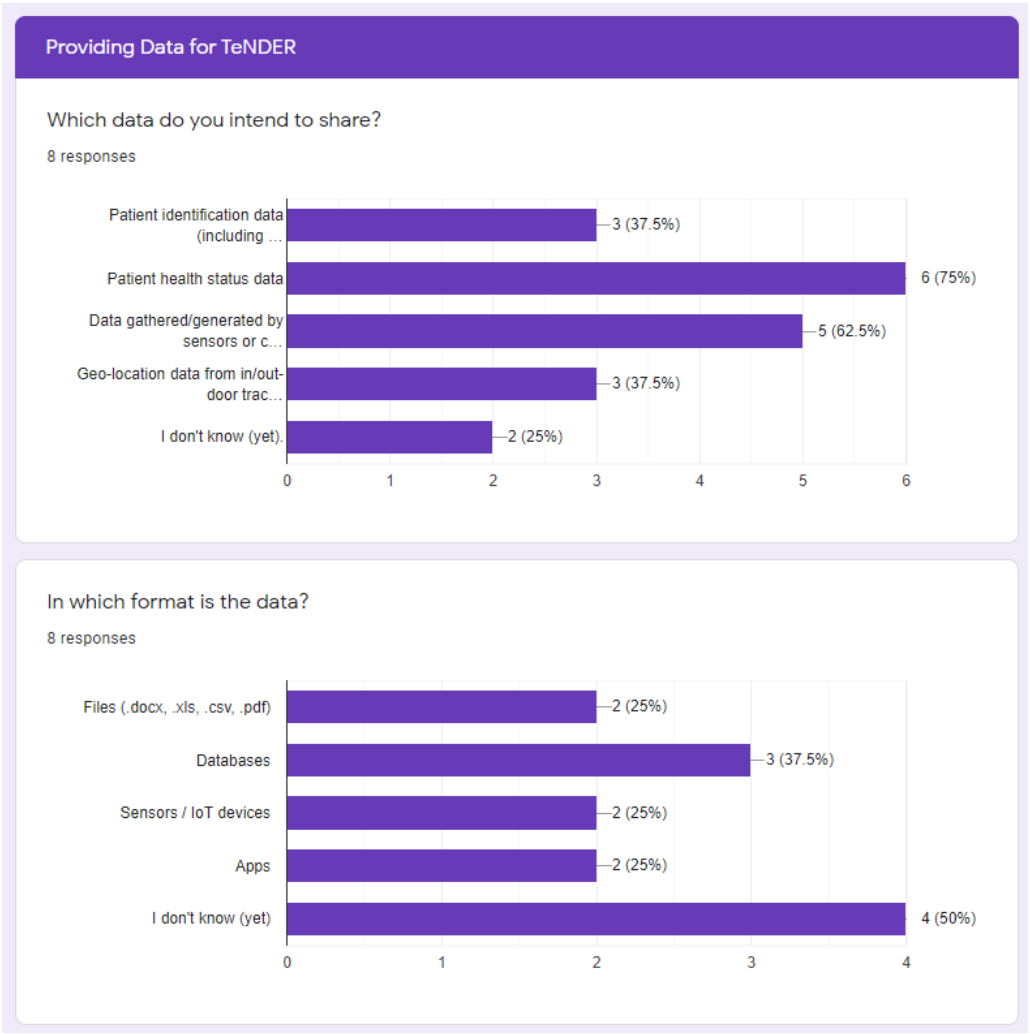


Figure 3 – Types and formats of data to be shared within TeNDER

As expected, and taking into account what had been defined in the project's description of action and in previous deliverables (concretely D1.1 and D2.1), the majority of the data to be shared by the pilot partners are linked to its patients and their healthcare professionals, namely the data that identifies them, their health status and quality of life assessment, the data that will be collected by the sensors and Internet of Things (IoT) devices installed in the pilot sites together with geolocation data for mood detection, activities tracking and pathway gathering. There is still a high uncertainty of how the data is stored, with some of the information (which is not provided by sensors) being available in databases, files, and applications. The survey then requested the partners' input about the preferred means of sharing the data and its compliance with EHRs.

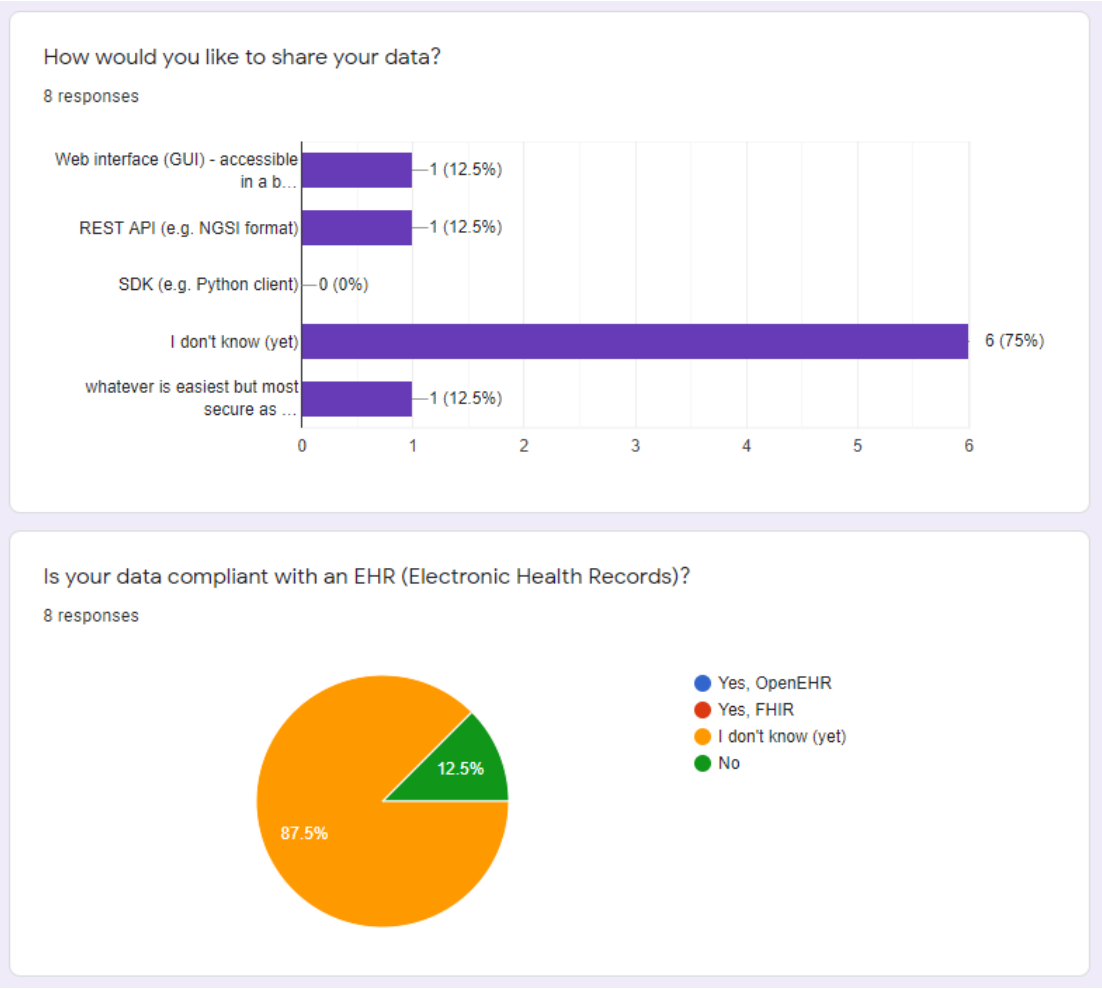


Figure 4 – Preferred ways to share data and compliance with EHR formats

However, the input was not very conclusive. Besides the information being scattered amongst different sources and formats, it appears not to be organised in standard information models or accessible via standard exchange protocols. Therefore, it creates an opportunity for TeNDER to design, implement and provide a tool that follows best practices from software architectures of reference projects in the healthcare domain, ensuring replicability as well as compliance with D1.1 requirements on privacy, ethical principles, data protection and security. The team thus gathered the state-of-the-art exchange protocols in the healthcare domain, as described in the following subchapter.

2.2 Information Gathering Standards

Interoperability is not possible without formal standards and specifications. Organisations such as Health Level Seven International[4] (HL7) and Personal Connected Health Alliance[5] (PCHAlliance) help towards the delivery of standards-based, open specifications that can support the flow of data from the point of capture into EHRs in the same format and coded content, but getting consensus on systems requirements is also important.

With the purpose of guaranteeing the secure and free flow of data within the EU, the new European Interoperability Framework[6] (EIF) has been announced in 2017, providing guidance to public administrations on how to improve governance of their interoperability activities. Through a set of recommendations, the EIF recommends how to streamline processes supporting end-to-end digital services, establish cross-organisational relationships, and ensure that existing and new legislation do not compromise interoperability efforts, as shown in the figure below.

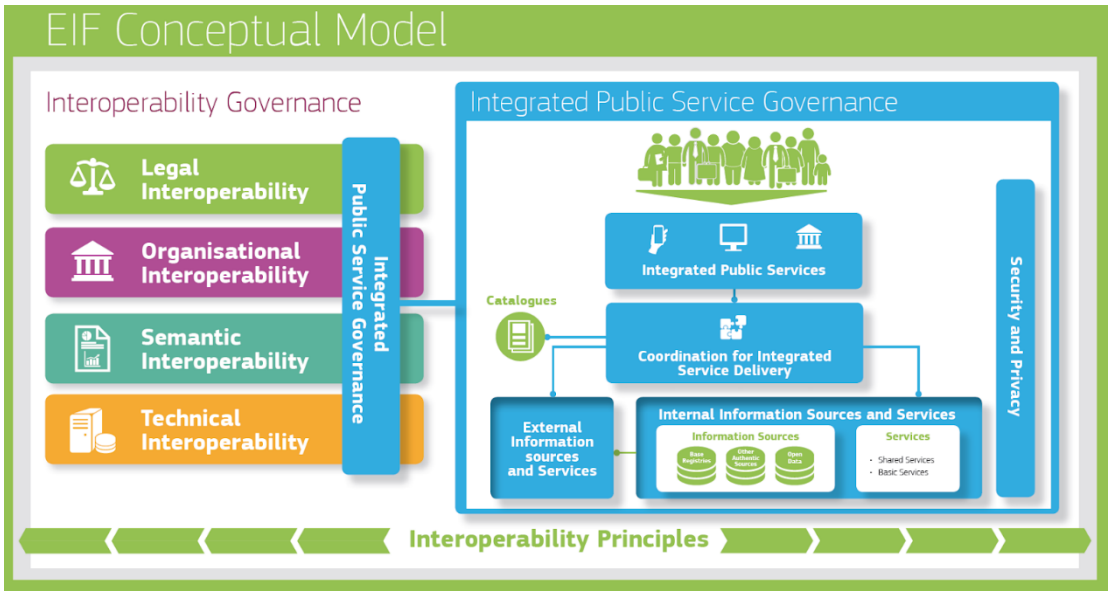


Figure 5 – The European Commission has adopted the new European Interoperability Framework (EIF) which will help European public administrations to coordinate their digitisation efforts when delivering public services.

The eHealth domain in Europe uses the refined eHealth European Interoperability Framework (ReEIF) as the common framework for managing interoperability in the context of the eHealth Digital Services Infrastructure supported under the Connecting Europe Facility Programme (CEF). Handling digitisation in public administrations in a coordinated way ensures that the public sector is not only digital but also interoperable, allowing them to save time, reduce costs, increase transparency, and improve the quality of services that they offer to citizens and businesses. This EU framework helps Member States to follow a common approach when making their public services available online, also across countries and policy areas, which contribute to reducing bureaucracy for people and businesses.

[4] Health Level Seven International – https://www.hl7.org/index.cfm
[5] Personal Connected Health Alliance – https://www.pchalliance.org/
[6] The New European Interoperability Framework – https://ec.europa.eu/isa2/eif

In 2019, the European Commission (EC) has adopted a recommendation on a European Electronic Health Record exchange format[7], to facilitate cross-border access to EHRs, while ensuring the highest levels of security and data protection. The recommendation proposes a set of common technical specifications for the transfer of health data in chosen health information domains such as Patient Summaries and ePrescriptions, but also laboratory test, images and hospital discharge reports and the further elaboration of the exchange format through a joint coordination process.

The EC states in its recommendation that "*Digitising health records, and creating systems that enable them to be securely accessed by citizens and securely shared within and between the different actors in the health system (patients, their clinical teams in the community and hospital facilities) is an important step towards integrating digital technologies into health and care approaches. That integration requires electronic health records, to be interoperable across the Union whereas currently many of the formats and standards in electronic health record systems – that are information systems for recording, retrieving and managing electronic health records – used across the Union are incompatible*".

Moreover, according to the same report, the lack of interoperability regarding electronic health records leads to *"fragmentation and a lower quality of cross-border healthcare provision*", having the EC identified *"specific 'Integrating the Healthcare Enterprise' (IHE) profiles listed in the Annex to Commission Decision (EU) 2015/130211,12 with the potential to increase interoperability of eHealth services and applications to the benefit of citizens and the healthcare professional community and to be eligible for referencing in public procurement. Those profiles provide detailed specifications for different layers of interoperability. Some of those profiles are already used to address specific business requirements in the eHealth Digital Service Infrastructure ('eHDSI')*".

IHE[8] is an initiative by healthcare professionals and industry to improve the way computer systems in healthcare share information. IHE promotes the coordinated use of established standards such as DICOM (Digital Imaging and Communications in Medicine) and HL7 to address specific clinical needs in support of optimal patient care. Systems developed in accordance with IHE communicate with one another better, are easier to implement, and enable care providers to use information more effectively. IHE is not a standard, although it supports the use of existing standards in an integrated manner, defining configuration choices. When clarifications or extensions to existing standards are necessary, IHE refers recommendations to the relevant standards bodies, therefore being an implementation framework, not a standard. Conformance claims must be made in direct reference to specific standards but may state that the products they describe are "implemented in accordance with the IHE technical framework" or "in compliance with the IHE technical framework."

Concerning the Electronic Health Record systems in Member States, the EC advocates that "*Member States should use the tools and building blocks provided by the eHealth Digital Services Infrastructure supported under the Connecting Europe Facility Programme and refer*

[7] Recommendation on a European Electronic Health Record exchange format – https://ec.europa.eu/digital-single-market/en/news/recommendation-european-electronic-health-record-exchange-format

[8] Integrating the Healthcare Enterprise (IHE) – https://www.ihe.net

to the Refined eHealth European Interoperability Framework[9] *as the common framework for managing interoperability in the eHealth domain*". A DSI (Digital Service Infrastructure) describes solutions that support the implementation of EU-wide projects, providing trans-European interoperable services which are composed of core service platforms and generic services, known as Building Blocks. Building Blocks are basic digital service infrastructures, which are key enablers to be reused in more complex digital services (cf. image below).

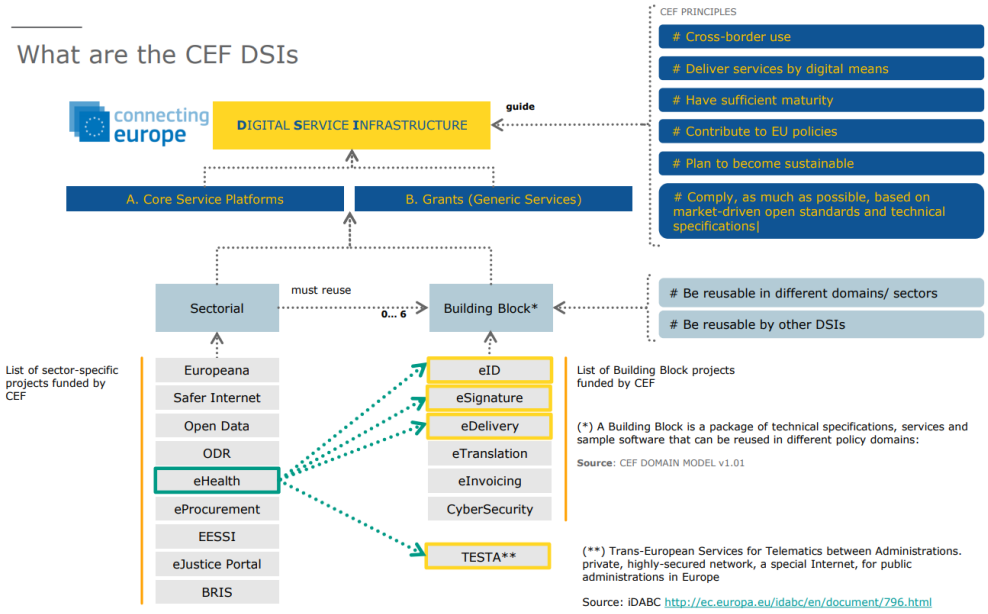


Figure 6 – CEF Digital Service Infrastructures (DSIs).

ReEIF is based upon the output of the Antilope project[10] (and specifically deliverable D1.1[11]), which took the eHealth European Interoperability Framework (eEIF) as a starting point.

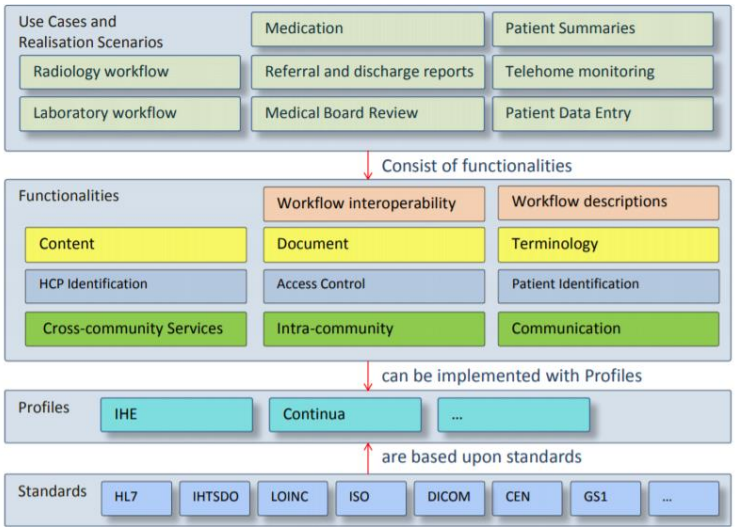


Figure 7 – Antilope Use Cases and Standards.

[9] eHealth Network Refined eHealth European Interoperability Framework – https://ec.europa.eu/health/sites/health/files/ehealth/docs/ev_20151123_co03_en.pdf
[10] Antilope Project – https://www.antilope-project.eu/front/index.html
[11] Refinement of Antilope Use Cases (D1.1) – http://www.antilope-project.eu/wp-content/uploads/2013/05/D1.1-Refinement_of_Antilope_Use_Cases_v1.2.pdf

Member States have taken important steps to foster interoperability with the support of the Commission, through the activities of the eHealth Network established under Article 14 of Directive 2011/24/EU of the European Parliament and of the Council (also known as the cross-border healthcare directive). The eHealth Network Guidelines on "*an interoperable eco-system for digital health and investment programmes for a new/updated generation of digital infrastructure in Europe*"[12] provides a list of technical specifications, standards and protocols to be used for already existing use cases, comprising specifications for:

1. Health information domains to be exchanged;
2. Interoperability specifications;
3. Cross-border exchange profiles.

The recommended interoperability specifications for content structuring and representation is presented in the following image:

Health information domains	**Clinical information for cross-border exchange**	**Content representation for cross-border exchange**
Patient Summary	Structured according to the provisions in the "GUIDELINE on the electronic exchange of health data under Cross-Border Directive 2011/24/EU Release 2 – Patient Summary for unscheduled care" adopted by the eHealth Network on 21 November 2016[5]	Health Level Seven (HL7) Clinical Document Architecture (CDA) Release 2[6] *Level 3 and Level 1 (PDF[7]/A)*
ePrescription/eDispensation	Structured according to the provisions in the "GUIDELINE on the electronic exchange of health data under Cross-Border Directive 2011/24/EU Release 2 – ePrescriptions and eDispensations" adopted by the eHealth Network on 21 November 2016[8]	Health Level Seven (HL7) Clinical Document Architecture (CDA) Release 2 *Level 3 and Level 1 (PDF[7]/A)*

Figure 8 – eHealth Network adopted guidelines for content structuring and representation for health information domains.

According to the eHealth Network guidelines, HL7 clearly stands out as a reference. HL7 and its members provide a framework (and related standards) for the exchange, integration, sharing, and retrieval of electronic health information. These standards define how information is packaged and communicated from one party to another, setting the language, structure and data types required for seamless integration between systems. HL7 standards support clinical practice and the management, delivery, and evaluation of health services,

[12] eHealth Network Guidelines to the EU Member States and the European Commission on an interoperable eco-system for digital health and investment programmes for a new/updated generation of digital infrastructure in Europe – https://ec.europa.eu/health/sites/health/files/ehealth/docs/ev_20190611_co922_en.pdf

and are recognised as the most (commonly) used in the world. One of the main categories of HL7 standards is Fast Health Interop Resources[13] (FHIR), an interoperability standard intended to facilitate the exchange of healthcare information between organisations. FHIR's premise is simple, lightweight and fast.

Consisting of two main parts (a content model in the form of 'resources', and a specification for the exchange of these resources in the form of real-time RESTful interfaces as well as messaging), FHIR can effectively be used as standalone specification for an electronic patient record system, data aggregation, exchange and reuse in acute care. More details on (European) EHRs are being collected under TeNDER's task T5.1 and will be published in D5.3 - First Report on the Health Record and Pathway repository.

Based on all these pieces of information, and considering the needs from TeNDER pilots and partners, the team has analysed the open-source community projects and solutions that implement FHIR[14], and defined the tool's architecture in the following section of the present deliverable.

2.3 Implementation and Architecture

The standard tool for information gathering is ultimately strongly linked to task T5.1, which is responsible for the implementation the European regulation for data exchanging, based on existing EHR systems, a database to store the medical profile will extend information available from physical, medical and behavioural activity, to allow the information securely flowing from patients, to the system and to the health professionals.

With this subsystem aiming at managing and organising patient information that is provided by a series of different low-level subsystems, HL7 FHIR has been chosen as the standard specification for data exchange. FHIR is a next generation standards framework created by HL7, and stands for "Fast Healthcare Interoperability Resources", leveraging the latest web standards and applying a tight focus on implementation. FHIR solutions are built from a set of modular components called "Resources", which can be easily assembled into working systems that solve real-world clinical and administrative problems, being suitable for use in a wide variety of contexts, such as mobile phone apps, cloud communications, EHR-based data sharing, server communication in large institutional healthcare providers, and much more.

In order to achieve this goal, an instance of HAPI FHIR Server[15] is being integrated, an open-source and complete implementation of the HL7 FHIR standard for healthcare interoperability in Java. HAPI has been designed to provide a flexible way of adding FHIR capability to applications, allowing different types of clients to connect to this server (cf. figure below).

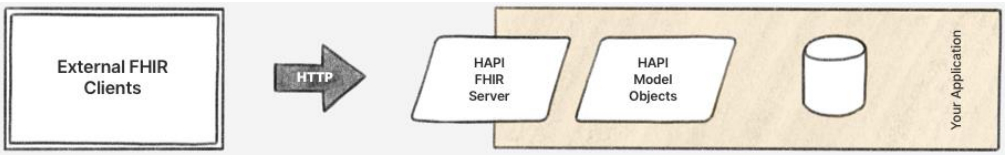


Figure 9 – EHR architecture regarding HAPI FHIR implementation.

[13] HL7 FHIR – http://hl7.org/fhir/
[14] Open-source FHIR Implementations – https://wiki.hl7.org/Open_Source_FHIR_implementations
[15] HAPI FHIR – https://hapifhir.io/

The HAPI Server's implementation of the FHIR standard provides an HTTP API to perform CRUD (create, read, update and delete) operations on the database, supporting different deployment schemes and relational databases. Initial tests are being done with HAPI's R4 version (since the latest is branded as unstable) and PostgreSQL v12.0 relational database (but others can be used, maintaining the structural integrity equal to the guidelines and examples provided in their documentation). The server has modules developed by the HAPI community that implement an assortment of functionalities and allow users to interact with the server with relative ease, which will support the other High-Level Subsystems (HLS) in TeNDER.

HAPI FHIR provides a built-in mechanism for connecting to FHIR REST servers. The HAPI RESTful client is designed to be easy to set up and to allow strong compile-time type checking wherever possible. A client has been set up as a proof of concept, using Java with Spring Framework, configured to use Apache Tomcat applicational server, and organised as a Model-View-Controller (MVC) pattern. At the moment of writing this deliverable, the proof of concept implemented a Controller with three endpoints described in the table below.

Table 2 - HAPI FHIR Sample Operations

METHOD	ENDPOINT	PARAMETERS	RESPONSE
GET	/patient/all	**name** - String - person name, surname **location** - A server defined search that may match any of the string fields in the Address, including line, city, district, state, country, postalCode, and/or text **orgId** - The id of the organisation that is the custodian of the patient record **gender** - gender of a patient **idRelatedPatient** - All patients linked to the given patient id **isActive** - Whether the patient record is active **phoneNumber** - A value in a phone contact **isDeceased** - This patient has been marked as deceased, or as a death date entered **email** - A value in an email contact **identifier** - A patient identifier (it can be a social security number, passport id, something unique!)	Bundle Resource (example: https://www.hl7.org/fhir/R4/bundle-example.json.html)
POST	/patient	**Patient Resource** (example: https://hl7.org/FHIR/patient-example.json.html)	Receives the same resource as it was entered
PUT	/patient/{id}	**id** – String	Receives the updated resource

Despite being a proof of concept compliant with eHealth standards and EC reference architectures and guidelines, there are other considerations to be included for this tool:

- The data sources are scattered in multiple formats like sensors, files and databases, so different clients need to be configured.
- The data to be collected not only contains private data, but it also reflects sensitive data, so explicit consent should be provided by patients and caregivers when submitting information to the platform.
- The overall architecture of the TeNDER platform is not yet concluded, and therefore this solution shall be adapted along the project's lifetime.

In the image below, an overview is provided about the data flow from pilots (left) to the core components defined for EHR gathering.

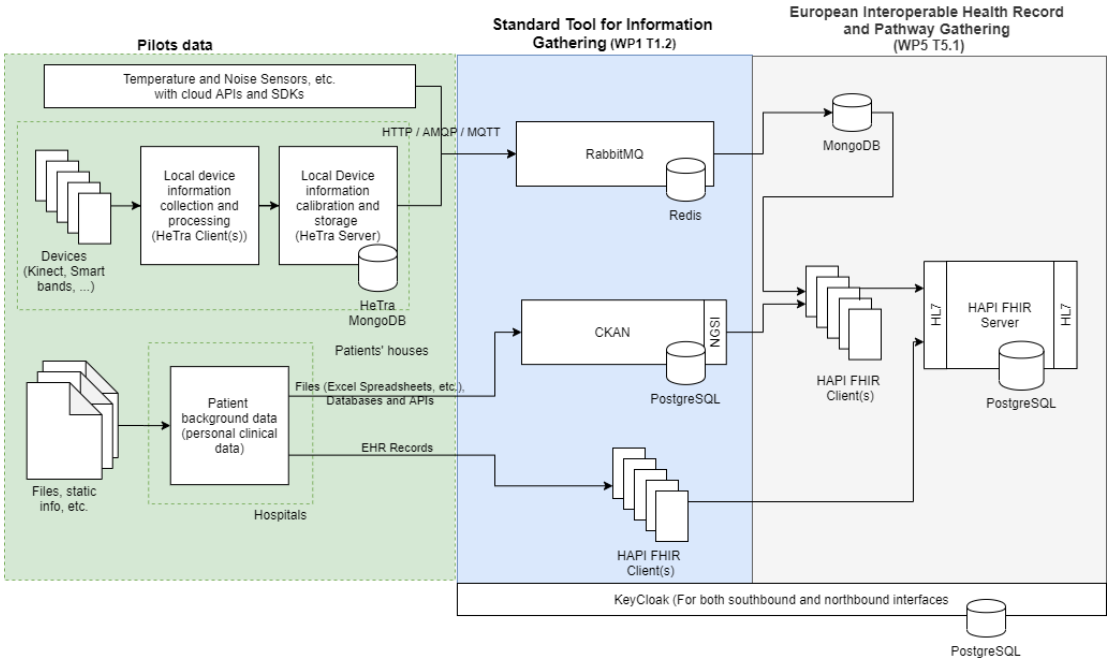


Figure 10 – Draft architecture for information gathering.

Based on the consortium experience from past and ongoing projects, three components are proposed for data collection (from bottom to top):

- HAPI FHIR Client(s) – for EHRs collection from the pilots.
- CKAN – for the collection and processing of files from the pilots.
- RabbitMQ – for the integration of real-time data from sensing devices.

Since the HAPI FHIR Client has already been described previously in the document, here we describe the other two modules, CKAN and RabbitMQ.

CKAN

According to CKAN's official documentation[16], CKAN is a tool for making open data websites, helping users to manage and publish collections of data, which is used by national and local governments, research institutions, and other organisations who collect a lot of data. With CKAN, once the data is published, users can use its faceted search features to browse and find the data they need, and preview it using maps, graphs and tables - whether they are developers, journalists, researchers, NGOs or citizens. In CKAN, data is published in units called "datasets", a parcel of data. Examples could be the health records for a hospital, the quality of life survey responses by caregivers, or temperature readings from weather stations. When users search for data, the results they see are individual datasets, each containing two things:

- Information or "metadata" about the data. For example, the title and publisher, date, what formats it is available in, what license it is released under, etc.
- A number of "resources", which hold the data itself.

[16] CKAN official documentation – https://docs.ckan.org

CKAN does not mind what format the data is in. A resource can be a CSV or Excel spreadsheet, XML file, PDF document, image file, linked data in RDF format, etc. CKAN can store the resource internally, or store it simply as a link, the resource itself being elsewhere on the web.

A dataset can contain any number of resources. For example, different resources might contain the data for different years, or they might contain the same data in different formats. This tool provides pilot partners with a user-friendly means to share information in an organised manner, as seen in the following images, collected from CKAN's website[17].

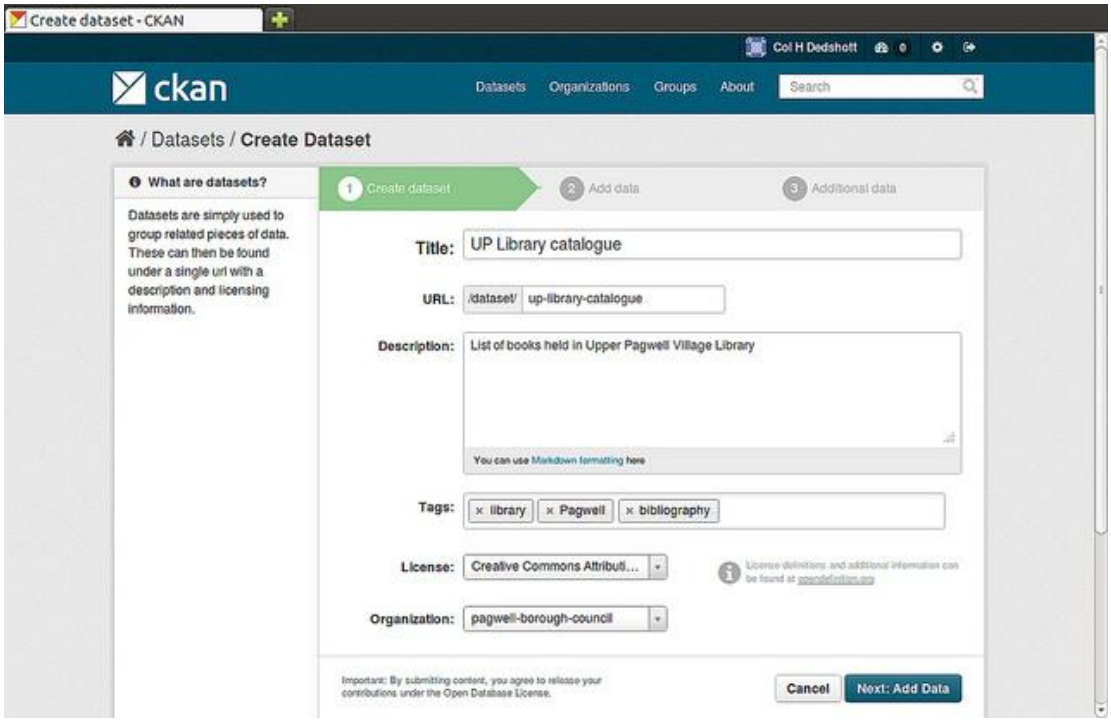


Figure 11 – Registering a dataset in CKAN.

Normally, each dataset is owned by an "organisation" with the CKAN instance being able to manage any number of organisations. Each organisation can have its own workflow and authorisations, allowing it to manage its own publishing process and its administrators to add individual users to it, with different roles depending on the level of authorisation needed. The users of an organisation can create a dataset owned by that organisation. In the default setup, this dataset is initially private, and visible only to other users in the same organisation. When it is ready for publication, it can be easily published although it may require a higher authorisation level within the organisation.

Datasets can be marked as public or private: the public ones are visible to everyone and the private datasets can only be seen by logged-in users who are members of the dataset's organisation. Private datasets are not shown in general dataset searches but are shown in searches within the organisation. Datasets cannot normally be created except within organisations despite being possible to set up CKAN to allow datasets not owned by any organisation that can be edited by any logged-in user. When creating a dataset, CKAN will ask for the following information about the data:

[17] CKAN User Guide – https://docs.ckan.org/en/2.8/user-guide.html

- Title – this title will be unique across CKAN, so it should be brief but specific. e.g. “Parkinson Madrid Caregivers” is better than “Caregivers”.
- Description – A longer description of the dataset can be added here, including information such as where the data is from and any information that people will need to know when using it.
- Tags – here one may add tags that will help people find the data and link it with other related data. Examples could be “Parkinson”, “health records”, “Madrid”.
- License – an important attribute so people know how they can use the data.
- Organisation - this drop-down enables the user to choose which organisation should own the dataset. In TeNDER, there will be different CKAN organisations for the different data owners.

After this step, users can then add or link the real data to the dataset being registered in CKAN (cf. image below).

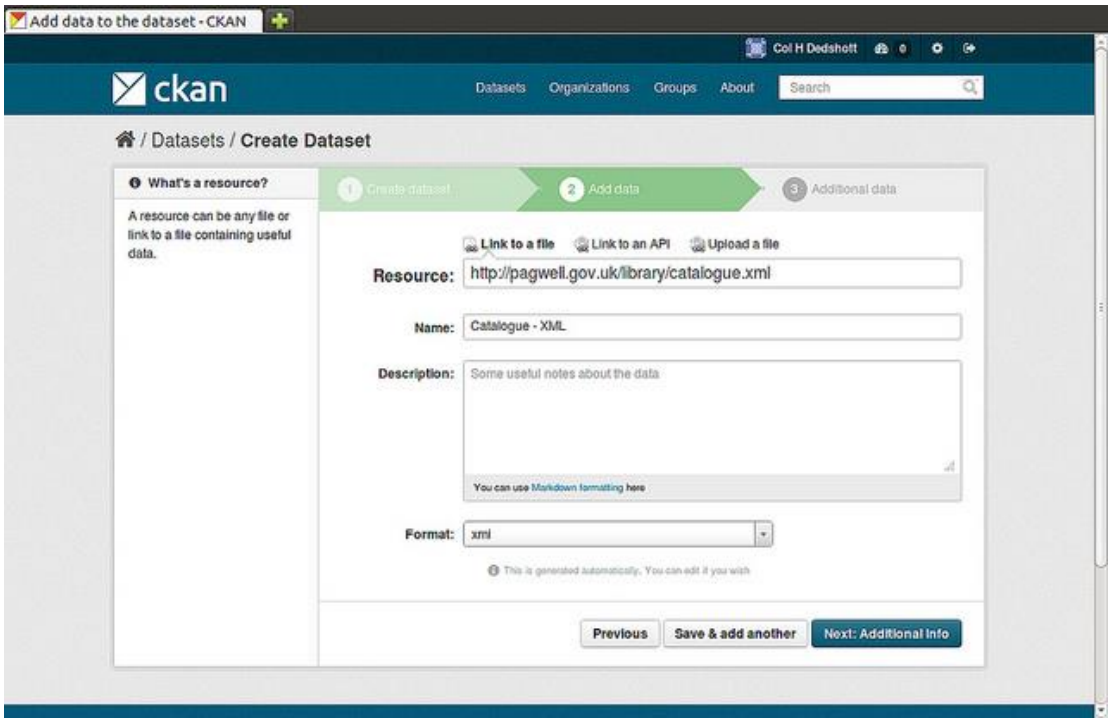


Figure 12 – Uploading a file to a dataset in CKAN.

Here one can add one or more “resources” which contain the data for this dataset, choosing a file or a link for your data resource and select the appropriate choice at the top of the screen. If one is giving CKAN a link to the data, like http://example.com/mydata.csv, then the “Link to a file” or “Link to an API” should be selected, while if the data to be added to CKAN is in a file on a computer, the other option should be selected. The rest of the information requested on the page is not required by CKAN, but it is good practice to add it, so other users can easily understand what the information is about:

- Name – a name for this resource, e.g. “Parkinson patients background, CSV”. Different resources in the dataset should have different names.
- Description – a short description of the resource.
- Format – the file format of the resource, e.g. CSV (comma-separated values), XLS, JSON, PDF, etc., so CKAN can properly render it in the dataset resources page.

A dataset can have multiple resources (files or links), so users can select the "Save & add another" button and add more resources to the created dataset.

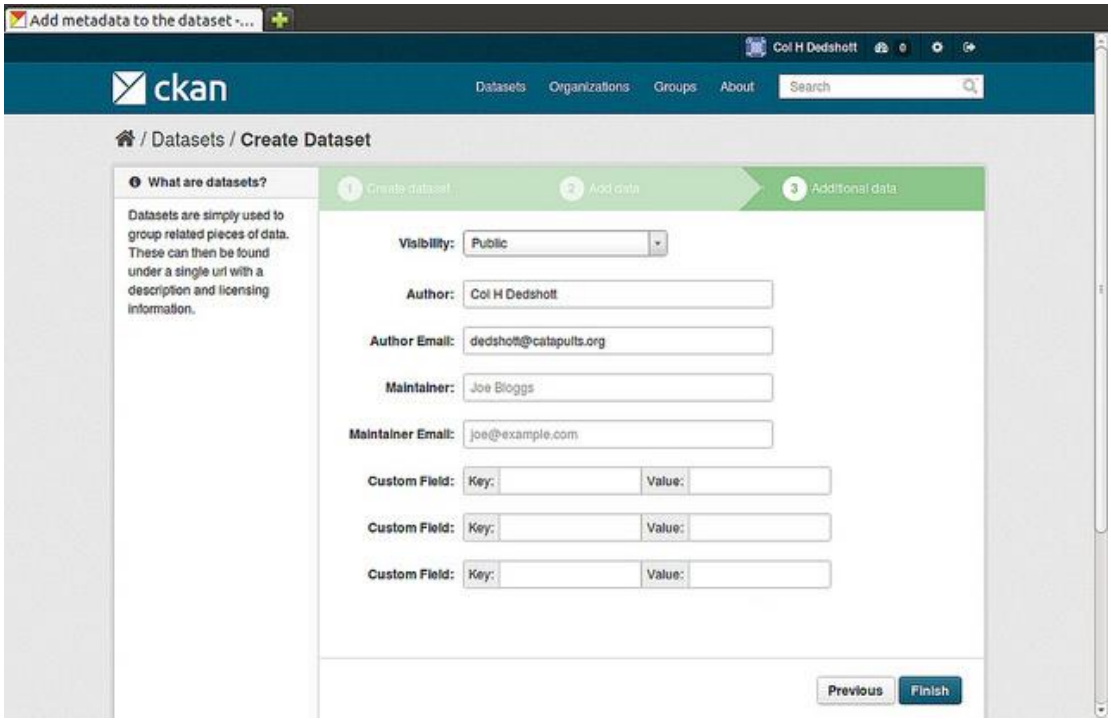


Figure 13 – Defining the visibility of a dataset in CKAN.

In the last stages of the dataset creation form, CKAN requests additional data for visibility and traceability purposes:

- Visibility – a Public dataset is public and can be seen by any user of the site. A Private dataset can only be seen by members of the organisation owning the dataset and will not show up in searches by other users.
- Author – the name of the person or organisation responsible for producing the data.
- Author e-mail – an e-mail address for the author, to which queries about the data should be sent.
- Maintainer / maintainer e-mail – if necessary, details for a second person responsible for the data.
- Custom fields – if one wants the dataset to have another field, one can add the field name and value here. e.g. "Year of publication". If there is an extra field that is needed for many datasets, CKAN administrators can change the default schema and dataset forms and enable custom fields for all the datasets.

CKAN then provides an API[18] and extensions[19] that enable data access for developers of applications on top, which is fully aligned with TeNDER's architecture and vision with the High-Level Subsystems, which will be provided in a later stage as Deliverable D2.3. Having been adopted as a FIWARE Generic Enabler, it can also be enriched with the NGSI open standard and FIWARE-based plugins[20].

[18] CKAN API Guide – https://docs.ckan.org/en/2.8/api/index.html
[19] CKAN extensions guide – https://docs.ckan.org/en/2.8/extensions/index.html
[20] FIWARE-CKAN-Extensions – https://fiware-ckan-extensions.readthedocs.io/en/latest/

RabbitMQ

RabbitMQ[21] is the most widely deployed open-source message broker, a lightweight and easy to deploy component, which supports multiple messaging protocols. RabbitMQ can be deployed in distributed and federated configurations to meet high-scale, high-availability requirements and runs on many operating systems and cloud environments, providing a wide range of developer tools for the most popular languages. RabbitMQ is a Message Broker that implements Advanced Message Queue Protocol (AMQP), that helps applications to communicate with each other, when one needs to scale their applications, thanks to exchanges and message queues (cf. image below).

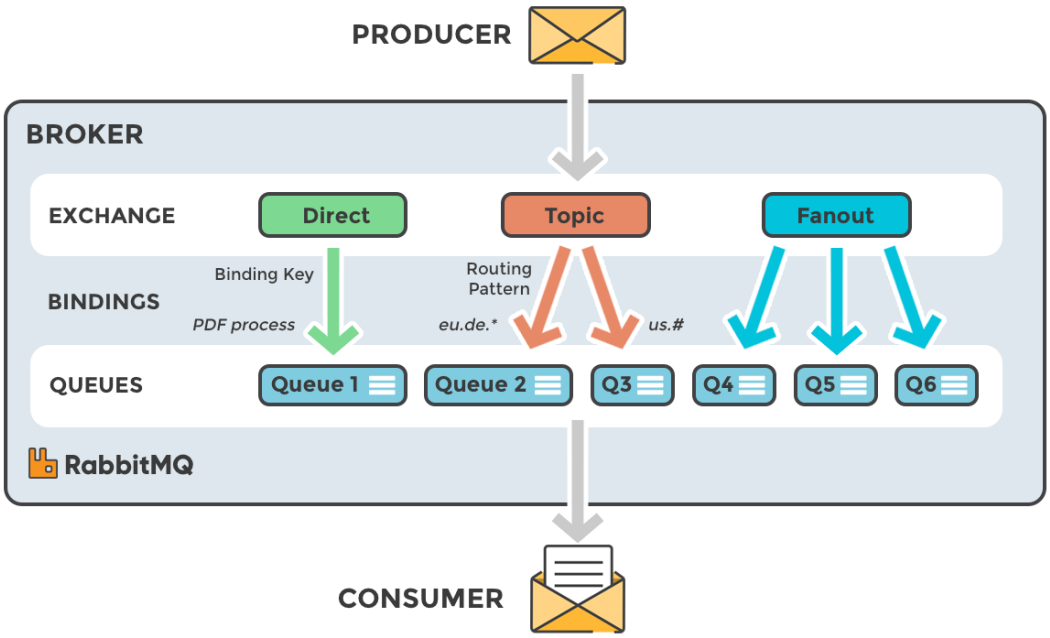


Figure 14 – RabbitMQ concept of data producers, data consumers, and different types of exchange, retrieved from https://www.cloudamqp.com/img/blog/exchanges-topic-fanout-direct.png.

Exchanges are entities where messages are sent, taking each message, and routing it into zero or more queues. The routing algorithm used depends on the exchange type and rules (called bindings). There are four types of logic that can be used:

- Direct Exchange – Messages are directed to a specific queue, based on the message routing key;
- Fanout Exchange – Messages are published to all queues that have the same routing key;
- Topic Exchange – Messages are published to all queues that have same routing key and routing pattern specified in the binding
- Headers Exchange – Headers exchanges ignore the routing key attribute, with the attributes used for routing being taken from the headers attribute instead.

Queues in RabbitMQ (or in AMQP) are entities similar to queues in other message- and task-queueing systems: they store messages that are consumed by applications. They share some properties with exchanges, but also have some additional properties, like name, durability,

[21] RabbitMQ – https://www.rabbitmq.com/

exclusivity and auto-deletion properties[22]. Bindings are rules that exchanges use (among other things) to route messages to queues. RabbitMQ documentation provides a nice analogy where a Queue is a destination (considering the European Commission office in Brussels), the Exchange would be the airport (e.g. Brussels International Airport) and the Bindings are the routes from the airport to the destination, where there can be zero or many ways to reach it.

The main reasons to having proposed RabbitMQ for this task is the flexibility and decoupling ability, i.e. separating the core components of the application (in this case the TeNDER platform) and improving its quality of Single Responsibility Principle[23]. Since the applications have been decoupled, it becomes flexible enough to connect different apps/services that written by different developers, teams and programming languages. The main features of the broker are therefore the following:

- Asynchronous Messaging – RabbitMQ supports multiple messaging protocols, highly-available message queuing, delivery acknowledgement, flexible routing to queues, multiple exchange types.
- Developer Experience – Users can develop cross-language messaging with different programming languages such as: Java, .NET, PHP, Python, JavaScript, Ruby, Go, and many others, as well as deploy them with BOSH, Chef, Docker and Puppet. There are multiple protocols and it supports many clients.
- Distributed Deployment – RabbitMQ can be deployed as clusters for high availability and throughput, federating across multiple availability zones and regions.
- Cloud Ready – Providing pluggable authentication and authorisation, it supports TLS and LDAP. Lightweight and easy to deploy in public and private clouds.
- Tools & Plugins – Being an extremely popular open-source component, it provides a diverse array of tools and plugins supporting continuous integration, operational metrics, and integration to other enterprise systems, a flexible plug-in approach for extending RabbitMQ functionality.
- Management & Monitoring – As a support to its integration and deployment, RabbitMQ comes with HTTP-API, command line tools, and UI for managing and monitoring the broker (tracing).

According to its documentation[24], clients communicate with RabbitMQ over the network, with all the protocols supported by the broker being TCP-based. The broker (and the operating system it is running on) is configurable and can be adapted to different TCP and IP operations, and application-level protocols such as TLS. Regarding the protocols[25] it uses to distribute information in a secure manner, RabbitMQ supports several messaging protocols, directly and through the usage of plugins, from AMQP to HTTP (and WebSockets), not leaving MQTT or STOMP behind.

This will enable for TeNDER to easily integrate with new and existing devices, sensors or dynamic data sources through standard, safe and reliable interfaces, enabling the platform to scale both horizontally as vertically (regarding data sources and platform scalability).

[22] AMQP concepts – https://www.rabbitmq.com/tutorials/amqp-concepts.html
[23] The Single Responsibility Principle – https://blog.cleancoder.com/uncle-bob/2014/05/08/SingleReponsibilityPrinciple.html
[24] Networking and RabbitMQ – https://www.rabbitmq.com/networking.html
[25] Which protocols does RabbitMQ support? – https://www.rabbitmq.com/protocols.html

3 CONCLUSIONS

The present deliverable reports on a standard tool for information gathering, based on state-of-the-art reference architectures and technologies, best practices from previous projects and EU initiatives and standards. It should be understood as a draft proposal, and not be considered final, since it is prior to D2.3 (that provides the overall architecture) and some of the datasets have not yet been validated or specified at the moment of writing.

Being a micro-services architecture, it can be altered or improved in a later stage, with the main goal for now targeting the validation by pilots and partners and the usage as a baseline in the architecture of the TeNDER platform.

Part of the work to be done within WP1 (and WP2) is the technical integration of the consent forms and procedure for data sharing and human participation in research, delivered by VUB in D10.2, D10.3 and D10.8, within all the tools specified here, so as to ensure that all end-users give consent before sharing private and sensitive information with the TeNDER ecosystem.

There are plenty of standard references in the European ecosystem, being a big priority now the cross-border data exchange. Although TeNDER is planning multiple pilots in different regions (from different member states) the challenge faced is not the same that the EC is fostering within CEF, but the team will ensure that the best practices and lessons learned are both adopted and promoted at a European level.