

Advanced Interoperability Techniques: Structure Mapping Service in CrowdHEALTH Project

Santiago Aso Lete¹,
Carlos Caverio¹, Andriana
Magdalinou², John Mantas²,
Lydia Montandon¹

¹ATOS, Madrid, Spain

²European Federation for Medical
Informatics, Lausanne, Switzerland

Corresponding author: Santiago Aso Lete, ATOS,
Spain. E-mail: santiago.aso@atos.net. ORCID
ID: <http://www.orcid.org/0000-0001-7019-7214>.

doi: 10.5455/aim.2020.28.52-57

ACTA INFORM MED. 2020 MAR 28(1): 52-57

Received: Dec 15, 2019

Accepted: Jan 20, 2020

ABSTRACT

Introduction: Due to healthcare information systems interoperability issues, health-related events may be missed. Numerous techniques based on medical standards and technologies have been recommended to address this challenge. Nevertheless, these techniques do not follow a holistic approach. **Aim:** The purpose of the paper is to present the structure mapping service which is developed in the context of the initial design of the interoperability solution of the CrowdHEALTH project. **Methods:** In the CrowdHEALTH project, subcomponents responsible for providing the appropriate interoperability solution have been created with first the rule engine for the implementation of the business logic. Another subcomponent responsible for creating and managing the knowledge related to the link that exists between information structures, or mappings between them, is the Structure Mapping Service. The envisioned approach for structure mapping is based on ontology alignment. Besides, the Terminology Service is responsible for providing a set of operations on medical terminologies used for the coding of medical knowledge, which fills the information structures. **Results:** Within the development of the Structure Mapping component, the Ontology alignment process will be automated. Two phases exist when using the structure mapping service: the training phase and the run-time phase. In the training phase ontologies are created and are aligned, and data maps between the use case (source) data sets and the HHR Manager classes (target) are formed. Afterward, these data maps can be stored in the knowledge base. In the runtime phase, queries can be made to the structure mapping component by the data converter. The structure mapping controller can then access the Knowledge Base to retrieve the appropriate mapping and transformations and provide these as a response to the query. **Conclusions:** The initial design of the advanced interoperability techniques is described with emphasis put to Structural Mapping service which can be easily deployed and seamlessly integrated into healthcare settings. This design will be assessed and further enriched during the project life circle.

Keywords: Interoperability, Data Structures, Project.

1. INTRODUCTION

Currently, the healthcare system faces various challenges ranging from excessive costs that could be avoided and information management issues to unnecessary Electronic Health Record (EHR) duplications and omission or unavailability of critical data (1). Such difficulties lie in the fact that providers are not able to exchange information and in most cases, they are still using obsolete formats. According to the IEEE (2) definition, interoperability is the “ability of a system or a

product to work with other systems or products without special effort on the part of the customer” made possible by the implementation of standards. The Semantic Health study (3) classifies the semantic interoperability of systems into those with local interoperability (within, e.g., hospitals or hospital networks) and those with regional, national and cross-border interoperability. Rector et al. (4) state that EHRs are “information models” and that “ontology” is the conceptualization of healthcare information (clin-

© 2020 Santiago Aso Lete, Carlos Caverio, Andriana Magdalinou, John Mantas, Lydia Montandon

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/4.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

ical, demographics, etc.). CrowdHEALTH project aims at processing datasets with structured and unstructured information coming from different Use Cases (5–9). Thus, there is a need to identify or map the entities and attributes from a proprietary source to a standard source specification and codify the information using standardized terminologies. For all these reasons, three components have been implemented: i) the Rule Engines which serve as interface to the rest of the components by orchestrating functionalities of other components as well as applying the rationale for transforming the clinical information, ii) the Structure Mapping which stores mapping rules from a given health information structure to another and iii) the Terminology Service which is responsible for carrying out different semantic operations on the available terminologies, like the retrieval of concepts, semantic inclusion, or translating between terminologies. The Structural Mapping sub-component will take advantage of well-established Ontology alignment approaches to perform the mapping between the schema/ model of an incoming document with the use-case specific target schema/model in the CrowdHEALTH platform. Ontology alignment approaches can be used for obtaining structural mappings between two different data models. Many tools and frameworks have been developed for aligning Ontologies that differ according to the degree of user intervention required to produce accurate mappings. In typical Ontology alignment approaches, data models or Ontologies are usually converted to a graph representation before being matched. In this context, aligning ontologies is sometimes referred to as “ontology matching”. The problem of Ontology Alignment has been tackled by trying to compute matching first and mapping (based on the matching) in an automatic fashion. Systems like DSSim, X-SOM or COMA++ (10) obtained high precision. There are also alignment tools such as PROMPT (11) for the Protégé framework. Data structures (12) are constructed by utilizing the related data values to create the linkages for bringing it together in multiple ways. Two different processes exist for either restructuring or reshaping the source data (13). Restructuring is achieved through the usage of the data relationships that exist in the data in terms of linkages, to reshape the structure to model these new relationships. In that case, new data relationships are applied, creating new meaning to the structure that is created. Also, new relationships’ linkages can be created through the comparison and formulation of the data relationships. On the other hand, reshaping aims at shifting the pieces of the structure around, without any kind of limits to the final structure. As a result, this allows the transformation of any structure into any other structure. Through Data transformation, the conversion of data from one format into a different one is achieved (14). Since data exists in various locations and multiple types, data transformation guarantees that data from one application or database can be used by other applications and databases. In case data has to be communicated, the data is extracted from the source, transformed into a different format, and is channeled to its

final destination. In the case of data integration, extract –transform –load (ETL) approaches are considered to be the leading processes (14). However, data may have to be merged, aggregated, enriched, or filtered according to different cases. Data mapping is considered to be the most important step in data transformation. The latter creates the relationship that exists between the data elements of two applications, providing ways of the transformation of data deriving from the data source, before it is finally provided to the target application. There are cases where data do not conform to a specific structure (e.g. email) and need data transformation to convert the provided information into data that can be used.

2. AIM

The aim of this paper is to present the envisioned approach for structure mapping is based on ontology alignment within the context of the CrowdHEALTH project.

3. METHODS

3.1. Structure Mapping Overall Approach

The envisioned approach for structure mapping is based on ontology alignment. The process involves running several matching operations either in series or parallel and then filtering the results of these matches to find an overall alignment. Once aligned a direct connection or mapping can be made between data elements in the source dataset and data elements in the target dataset. Within the development of the Structure Mapping component, the Ontology alignment process will be automated. Two phases exist when using the structure mapping service: the training phase and the runtime phase. In the training phase ontologies are created and are aligned, and data maps between the use case (source) data sets and the HHR Manager classes (target) are formed. Afterward, these data maps can be stored in the knowledge base. In the runtime phase, queries can be made to the structure mapping component by the data converter. The structure mapping controller can then access the Knowledge Base to retrieve the appropriate mapping and transformations and provide these as a response to the query.

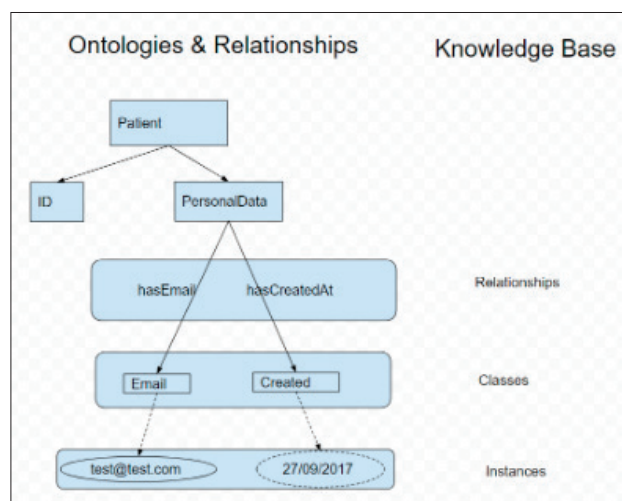


Figure 1. Sample ontology for a use-case

3.1.1. Training Phase

Before any live structure mapping, operation Ontologies are created for each use-case using the data models provided by each use case partner in the CrowdHEALTH project as per Figure 1. Creating the Ontologies means deriving the key elements, structures, and relationships and devolving this to triples of information. The other stream of work is to establish a target data model or Ontology-based on the HHR Manager classes defined in the CrowdHEALTH project. Once the ontology has been established for a use case and there is the ontology for the HHR Manager classes, the two ontologies can be aligned. Matches between Ontologies can be based on the relationships and dependencies between name structure and placement of instances.

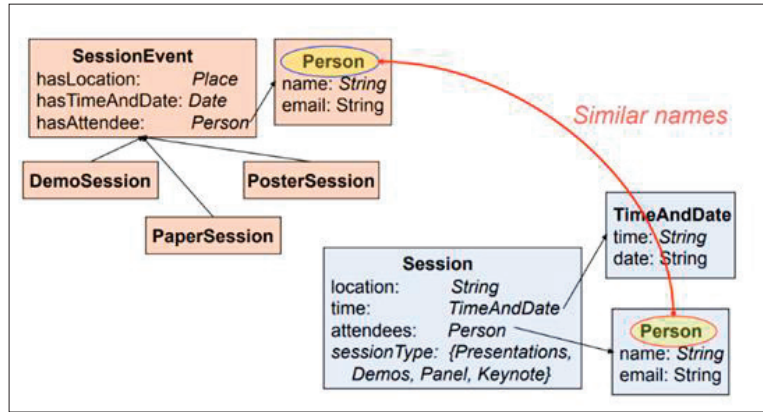


Figure 2. Similar names

Name similarity can be recognized by utilizing semantic interoperability approaches that aim at identifying similarities between the name of different elements –as shown in Figure 2. Structural similarity can be recognized when both Ontologies have common members or relationships as per Figure 3. Instance similarity can be recognized by examining instances of data from the two ontologies (Figure 4). Part of CrowdHEALTH’s vision is to form the best combination of established approaches to provide automatic alignment of use-case data models with the target HHR model.

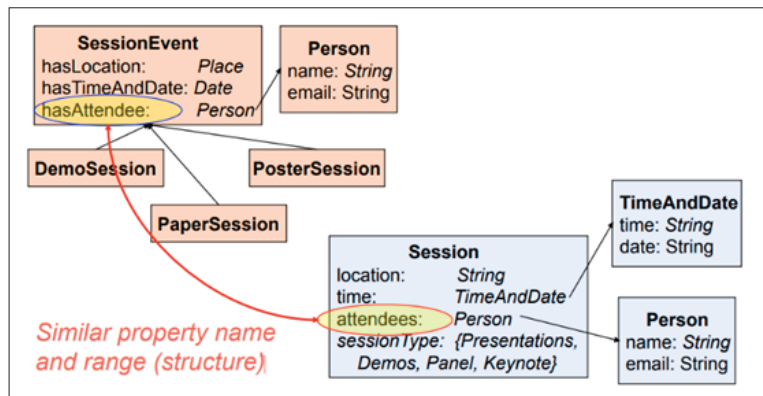


Figure 3. Instance similarity based on common data occurrence between two ontologies

An example pipeline for Ontology alignment is shown in Figure 5. Once the target HHR structure and Use Case data structures have been mapped through the Ontology alignment approach, the Structure Mapping component will be responsible for providing the rules or schemata describing the mapping of incoming data with the target data structure as shown in Figure 6. The mapping rules or schemata will be stored in the Knowledge Base.

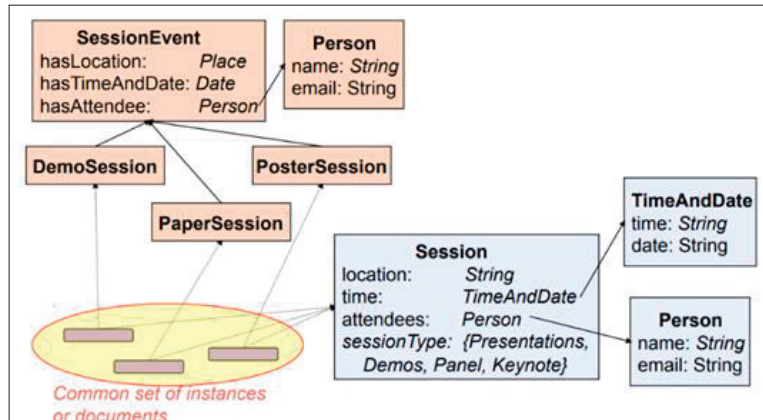


Figure 4. Common occurrence between two Ontologies can be identified through Ontology alignment

3.1.2. Runtime Operation

The runtime operation of the Structure Mapping component involves the implementation of Data Maps that processes incoming data structures from the use cases to be mapped to the structure of the HHR Manager class structure. The runtime operation of the process is covered in the following steps: Incoming data is received from the Ruler Engine and is converted into a common internal or use-case specific format by the structure mapping service. The incoming data is fed into a Data Map that is based on the Ontology alignment approach i.e. the Data Map aligns the use-case data model with the target HHR model as shown in Figure 7. If necessary, the outcome of the Data Mapping step is translated to create an outcome where the incoming data is mapped to the target HHR data

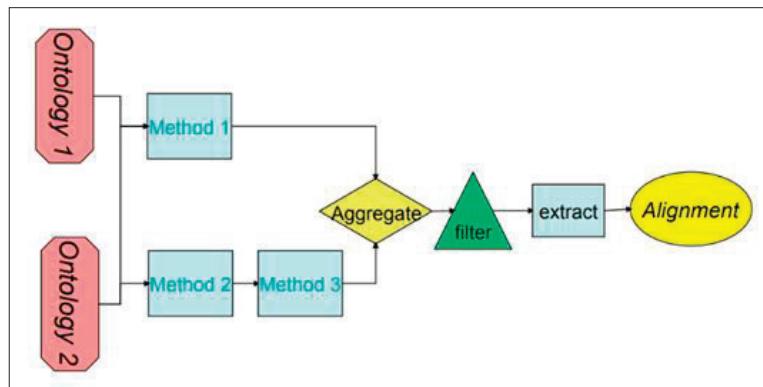


Figure 5. Example of Ontology alignment

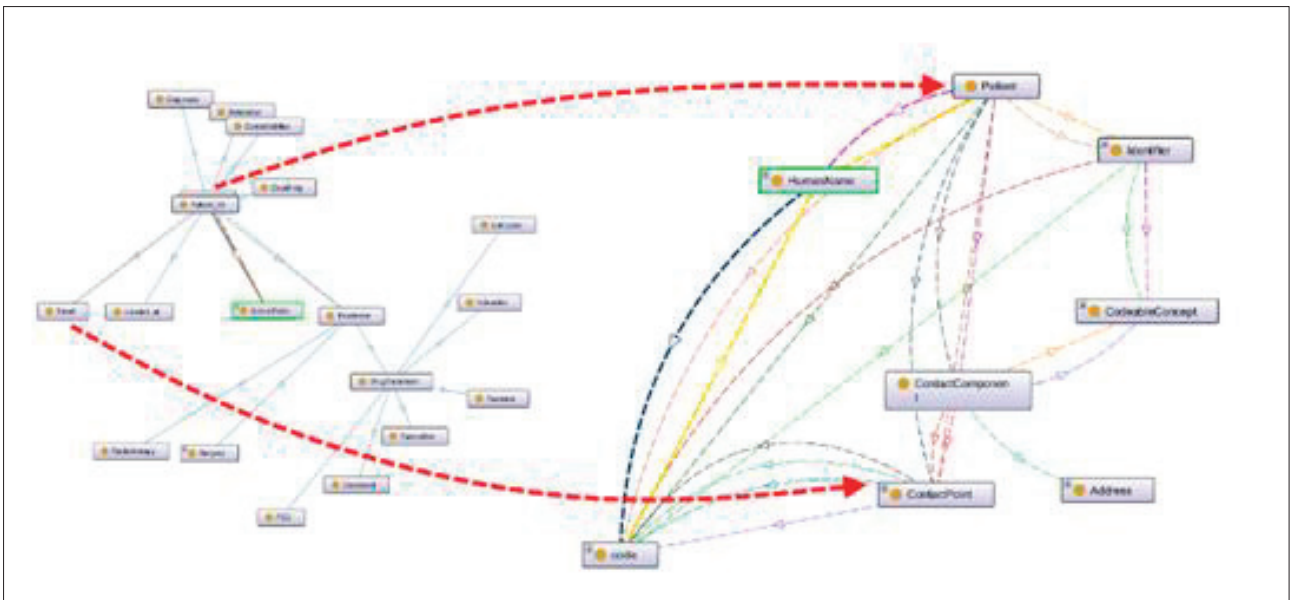


Figure 6. Sample, Ontology alignment approach

model. The terminology service is called if specific transformations are required (Figure 8). The outcome of the Structural Mapping sub-component is forwarded to the other components in the CrowdHEALTH platform. The Data Maps are updated (if needed) and stored for future use.

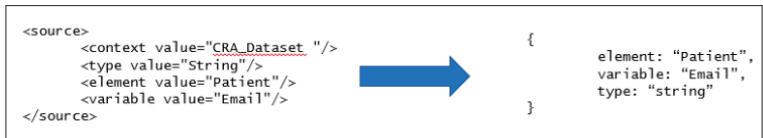


Figure 7. Data Maps will implement the Ontology Alignment approach to translate data from use case specific structure to target HHR structure

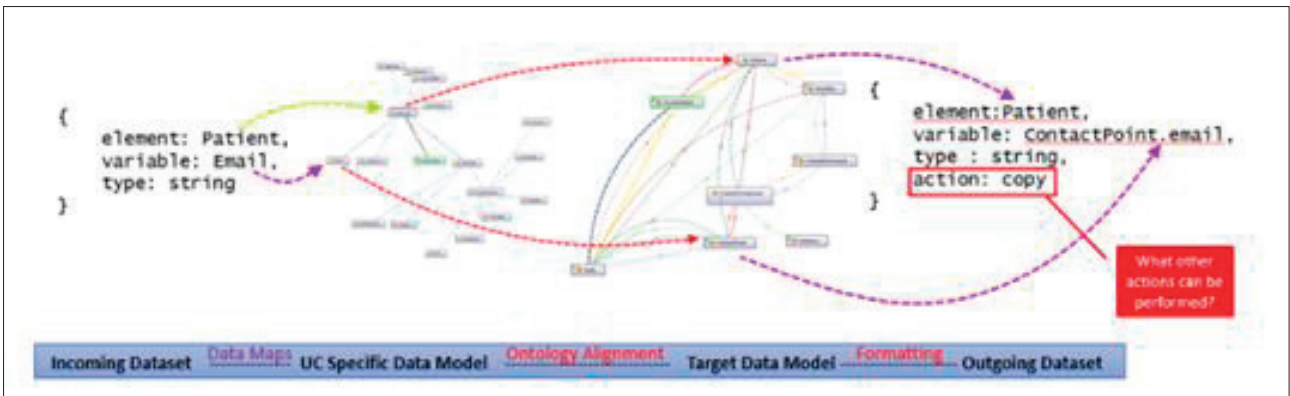


Figure 8. Transformation of data from initial to final stage

3.2. Structure Mapping internal architecture

The internal architecture of the Structure Mapping sub-component is illustrated in Figure 9. The Structure Mapping sub-component provides a Representational State Transfer (RESTful) Application Programming Interface (API) for the Data Converter to access the Structure Mapping service. In response to requests from the Data Converter, the Structure Mapping service returns mappings for incoming data, the mapping describes how the data elements in the various uses-cases can be transformed into the HHR Manager structure. It also returns for each element any required translation function to convert the data to the correct HHR format. The following modules deliver the functionality of the Structure Mapping service.

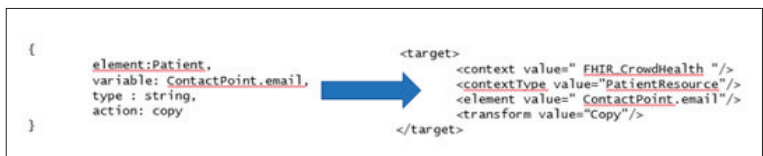


Figure 9. Internal architecture of the Structure mapping subcomponent

3.2.1. Interface Layer

The interface layer is designed to allow variations in the implementation of the actual Structure Mapping service. In all cases, the Interface Layer handles conversion to and from JavaScript Object Notation (JSON)/ Extensible Markup Language (XML) to one internal format of the structure mapping engine i.e. JSON.

3.2.2. Knowledge Layer

The Knowledge Layer consists of two elements: the Ontology Layer and the Knowledge Base. At training

time, the Ontology layer is actively aligning use case data set ontologies to the HHR Manager Class ontology, the result of these alignment operations are stored in the knowledge base. Once training is complete the Knowledge Base will contain maps that map use-case specific data models to the target HHR model. At run time the primary function of the ontology layer is to provide these mappings in response to requests passed through the structure mapping controller.

3.2.2.1. Ontology Layer

Several possibilities are available for creating Data Maps using Ontology Alignment approaches. However, the focus of this component is to create an automated or semi-autonomous technique for selecting and executing relevant Data Maps. The actual process of Ontology alignment involves several sub-components combined into a pipeline. During training the key elements will be transforming the use-case data to triplets, feeding these into a series of matches to compare with an established HHR model in the CrowdHEALTH, aggregating the results and filtering to produce an alignment with the possibility of iterating this process to refine results.

3.2.2.2. Knowledge Base

Once created, Data Maps are stored in the knowledge base and validated against existing use-case data. Tools can be used to provide visualization of the result as a graph: (Intermediate point between ontology layer and knowledge base). The purpose of the knowledge base is to provide a local data store to the Structure Mapping controller module, containing details of the relationships, classes, and instances identified by the Ontology Layer. A Triple Store database is developed using established technology such as RDFLib for Python or SPARQL. This is implemented as a standalone Docker container with interfaces to provide information in response to requests from the Structure Mapping Controller, work as a storage container for mappings from the Ontology Layer and offer a query language to process requests from the structure mapping controller.

3.2.3. The structure mapping controller

The Structure Mapping Controller is responsible for selecting an appropriate Data Map when being supplied with incoming data. The Controller analyzes the incoming data and accesses the Knowledge Layer to retrieve relevant Map, with links to the corresponding resource and entity in the target HHR model.

3.2.4. Error logging and sanity checking

Due to the fuzzy nature of the matching performed by the Structure Mapping Controller and the need for updates many actions are considered and included:

- Log structure mapping interactions,
- Check over a range of results for data anomalies,
- Provide a method for signaling error processing,
- Provide an update or retraining method for the Knowledgebase,

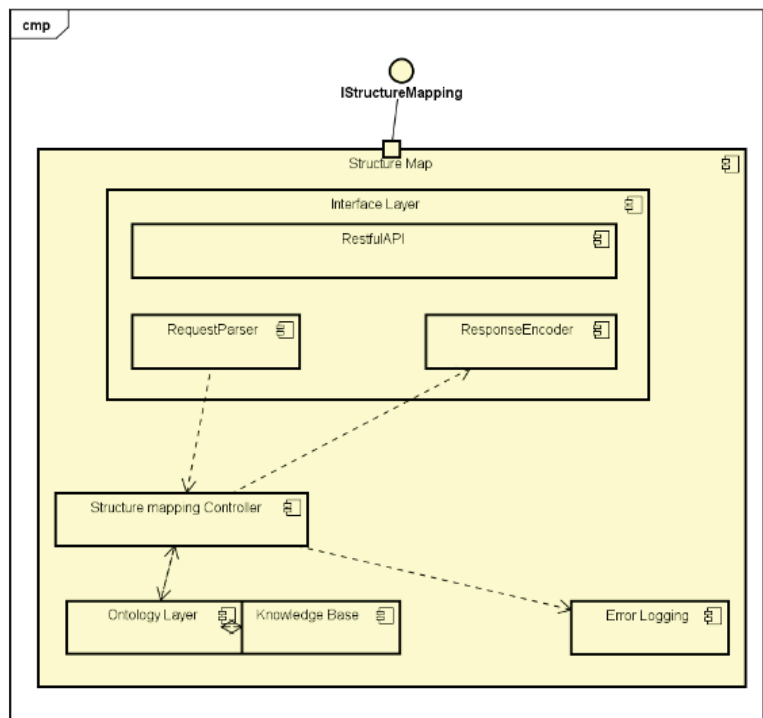


Figure 10. Process to implement and refine Data Maps

4. CONCLUSION

Provide graphing tools to enable the existing Knowledgebase to be examined.

Healthcare systems need to tackle interoperability issues and communicate in the same language to optimize patient outcomes and improve care delivery. The CrowdHEALTH project provides an infrastructure to transform the clinical information into processable data using eHealth standards and terminologies. The initial design of the advanced interoperability techniques was described with emphasis put to Structural Mapping service which can be easily deployed and seamlessly integrated into healthcare settings. This design will be assessed and further enriched during the project life cycle.

- **Acknowledgement:** CrowdHEALTH project is co-funded by the Horizon 2020 Programme of the European Commission Grant Agreement number: 727560 – Collective wisdom driving public health policies.
- **Author's contribution:** Each author gave substantial contribution in acquisition, analysis and data interpretation. Each author had a part in preparing article for drafting and revising it critically for important intellectual content. Each author gave final approval of the version to be published and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.
- **Conflict of interest:** None declared.
- **Financial support and sponsorship:** Nil.

REFERENCES

1. Mantas J. Future trends in Health Informatics – theoretical and practical. *Studies in Health Technology and Informatics*. 2004; 109: 114-127.
2. The Role of Standards in Engineering and Technology. Institute of Electrical and Electronics Engineers (IEEE). 2007. Available at: http://www.ieee.org/education_careers/education/stan-

- dards/standards_glossary.html.
3. European Commission. Semantic HEALTH Semantic Interoperability for better health and safer healthcare Semantic HEALTH Report. Information Society and Media Directorate-General. 2009.
 4. Rector A, Qamar R, Marley T. Binding Ontologies and Coding systems to Electronic Health Records and Messages. KR-MED Biomedical Ontology in Action. 2006. doi: 10.3233/AO-2009-0063.
 5. Kyriazis D, Autexier S, Boniface M, Engen V, Jimenez-Peris R, Jordan B. et al. The CrowHEALTH Project and the Hollistic Health Records: Collective Wisdom Driving Public Health Policies. *Acta Inform Med.* 2019 Dec; 27(5): 369-373. doi: 10.5455/aim.2019.27.369-373.
 6. Perakis K, Miltiadou D, De Nigro A, Torelli F, Montandon L, Mantas J. et al. Data Sources and Gateways: Design and Open Specification. *Acta Inform Med.* 2019 Dec; 27(5): 341-347. doi: 10.5455/aim.2019.27.341-347.
 7. Wajid U, Orton C, Mogdalinou A, Mantas J, Montandon L. Generating and Knowledge Framework: Design and Open Specification. *Acta Inform Med.* 2019 Dec; 27(5): 362-368. doi: 10.5455/aim.2019.27.362-368.
 8. Magdalinou A, Mantas J, Montandon L, Weber P, Gallos P. Disseminating research Outputs. The CrowdHEALTH Project. *Acta Inform Med.* 2019 Dec; 27(5): 348-355. doi: 10.5455/aim.2019.27.348-355.
 9. Malliaros S, Xenakis C, Moldovan G, Mantas J, Magdalinou A, Montandon L. The Intergrated Holistic Security and Privacy Framework Deployed in CrowdHEALTH Project. *Acta Inform Med.* 2019 Dec; 27(5): 333-340. doi: 10.5455/aim.2019.27.333-340.
 10. Rahm E, Arnold P, Do H. et al. COMA 3.0. University of Leipzig. Available at: <https://dbs.uni-leipzig.de/de/Research/coma.html>.
 11. University of Stanford. PROMPT. Available at: <https://protege-wiki.stanford.edu/wiki/PROMPT>.
 12. Introduction to Data Structures. Available at: <http://www.studytonight.com/data-structures/introduction-to-data-structures>.
 13. David M. Hierarchical Data Structure Transformation in SQL. *Database Journal.* 2010. Available at: <http://www.databasejournal.com/sqlc/article.php/3898591/Hierarchical-Data-Structure-Transformation-in-SQL.htm>.
 14. Data Transformation. Available at: <https://www.mulesoft.com/resources/esb/data-transformation>.
 15. Syncsort. ETL (Extract – Transform – Load). Available at: <https://www.syncsort.com/en/glossary/etl>