Clinical Process Management: A model-driven & tool-based proposal

Julián Alberto García García *julian.garcia@iwt2.org*
Web Engineering and Early Testing Group, University of Seville
Seville, Spain

María José Escalona Cuaresma *mjescalona@us.es*
Web Engineering and Early Testing Group, University of Seville
Seville, Spain

Alicia Martínez-García *alicia.martinez.exts@juntadeandalucia.es*
Technological Innovation Group, "Virgen del Rocío" University Hospital
Seville, Spain

Carlos Parra *carlos.parra.sspa@juntadeandalucia.es*
Technological Innovation Group, "Virgen del Rocío" University Hospital
Seville, Spain

Tomasz Wojdyński *tomwoj@wszib.edu.pl*
School of Banking and Management
Crcow, Poland

Abstract

In healthcare institutions it is important to define methodologies and management strategies in order to define, maintain and execute Healthcare Processes (HP) in a simple and effective manner. These needs are necessary because this kind of processes involve many people (software engineer, healthcare teams, and doctors, among others) and must also comply with lots of international clinical standards (such as ISO-EN 13606 or ISO/DIS 13940). This paper presents a formal demonstration of our proposal which is based on the Model-Driven paradigm in order to support modeling, deploying and executing HP. Our tool-based proposal allows reducing costs, improving quality and optimizing HP taking into account the Model-Driven paradigm advantages.
Keywords: Model-Driven Engineering; healthcare processes; health standards; metamodels; archetypes.

1. Introduction

According to M. Weske, *"A business process consists of a set of activities that are performed in coordination in an organizational and technical environment. These activities jointly realize a business goal. Each business process is enacted by a single organization, but it may interact with business processes performed by other organizations."* [1]. This paper assumes the previous definition, but it is focused on the business process management in the field of healthcare. In this environment, business processes are focused on improving the care of patients and they are usually named as Healthcare Process (HP).

Although in general, a proper business processes management can contribute to reduce costs and improve quality as well as optimizing all kind of business processes, it is necessary to provide support tools in order to guarantee quality and applicability of these processes in real environments. However, defining Healthcare Processes Support Systems (HPSS) is not always easy task within healthcare context because there are many people with different points of view of the same system (such as Healthcare Experts (HE), Information Technology (IT) teams, clinical professionals, among others). In addition, it is often necessary to take into account lots of international clinical standards (such as ISO-EN 13606 [2] or ISO/DIS 13940 ContSys [3], among others) which establish guidelines and policies to define clinical data and possible interdependencies between them.

Moreover, the definition of HPSS requires a complex phase of requirements engineering where IT teams must implement HPSS following specific methodologies and international clinical standards (as those mentioned above), but it is required to define appropriate mechanisms to assure that the implementation of this HPSS matches with its definition. This task is sometimes difficult because the IT team who implements HPSS is not always the same who defines this HPSS [4]. This may provoke inconsistencies and traceability problems. In addition, if teams do not know clinical standards properly, it can be very expensive to extend, maintain evolve and adapt these systems.

In this paper, we present a demonstration of our proposal based on the Model-Driven Engineering (MDE) [5] paradigm. Our proposal is also supported by tools in order to model, deploy and execute Healthcare Processes within healthcare institutions. We have considered MDE because it helps us to manage the conceptual complexity of clinical process and allows defining clinical concepts and relationships between them in a general manner. In addition, a concrete syntax to represent these concepts in real environments has been defined to offer a suitable way to instance each concept [8].

Our proposal has been also designed taking into account three international standards which are important in healthcare environments because they are related to business processes and clinical data.

On the one hand, we take into account the ISO/IEC TR 24744:2007 standard which presents uniformity guidelines to define process models.

On the other hand, we take into account the CEN/ISO EN13606 standard which defines rigorous and stable information architecture for transferring part or all of the Electronic Health Record (EHR) of a single subject of care (patient) among EHR systems, or between EHR systems and a centralized EHR data repository. It may also be used for EHR communication between an EHR repository and clinical applications (such as decision support components).

Finally, we take into account the UNE-EN 13940 standard which defines the generic concepts needed to achieve continuity of care, which is a relevant aspect in relation to quality and safety in healthcare, with semantic interoperability, a fundamental requirement for continuity of care.

After this introduction, this paper is structured as follows. Section 2 describes briefly our proposal and how this one has been apply in the eHealth platform of the Andalusian Regional Ministry of Health and Social Welfare. Section 3 describes our supporting tool and explains a real successful case performed in collaboration with “Virgen del Rocío” University Hospital (VRUH) [7]. Finally, Section 4 states conclusions and introduces future lines of research.

2. Approach to clinical process management

Our proposal aims to (*i*) integrate international standards during the definition of HPSS, (*ii*) assure the real traceability with the definition of HP and the more effective way to implement HP as a software solution, (*iii*) make easier the communication between clinical experts and IT teams in order to reduce the cost and the risk of this communication, and (*iv*) improve the HP maintenance in an easy, effective and efficient manner. For this purpose, we propose a five-step methodology based on the MDE:

1. Modelling HP conforms to a clinical process metamodel which takes into account the ISO/IEC TR 24744 standard.
2. Modelling clinical data through constraints and relationship between them. This step takes into account international standards such as ISO-EN 13606; ISO 21090 [10], which harmonizes the definition data types for information technology related to

health; or ISO/DIS 13940 ContSys, which establishes concepts associated with HP and their relationships based on continuity of care concepts.

3. Establishing relationship between clinical process and clinical data.
4. Generating initial software code to allows deploying these models onto eHealth platforms.
5. Adapting the generated code to include specific features of the concrete eHealth platform.

Figure 1 shows schematically each step which are supported by a whole theoretical framework.

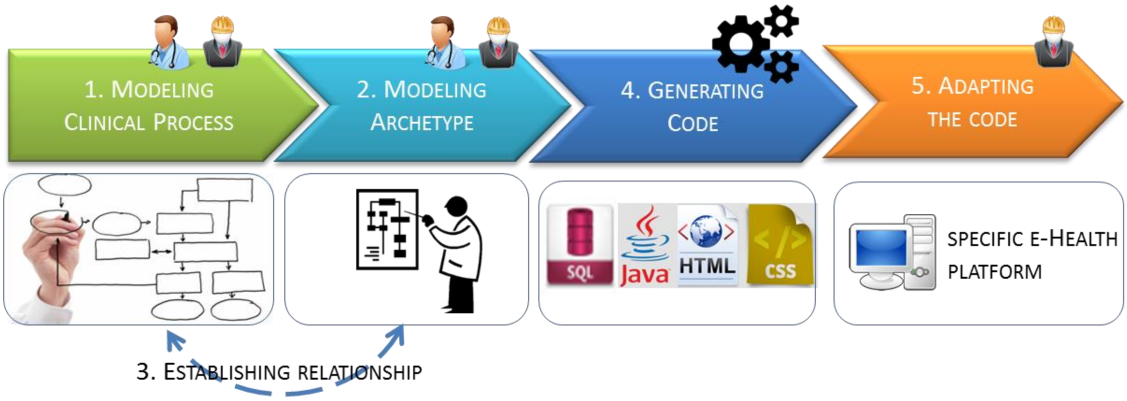


Figure 1. Clinical process management methodology

The **first three steps** are supported by two metamodels which have been defined using UML (Unified Modelling Languages) [12] class diagram. These steps are carried out by Process Engineers and Clinical Professionals in order to (*i*) capture all requirements of the clinical process and (*ii*) define data structures of clinical information.

On the one hand, we have defined a clinical process metamodel to achieve the first goal. This metamodel is an adaptation of the model-based process modelling language defined within the PLM$_4$BS (Process Lifecycle Management for Software-Business) framework. We have decided to reuse this metamodel in clinical environments because we have achieved very good results during its application in other real environments (software processes, public administration processes, etc.). This metamodel has been defined in compliance with the guidelines and recommendations for the ISO/IEC TR 24744 standard and also taking into account the business process metamodel defined in [11]. With this metamodel, user can be able to define processes of considerable complexity in a simple and effective manner.

Figure 2 shows a general vision of our process metamodel [11]. This metamodel is generic and it can be applied in any business context. For instance, it can be applied in Health environments, but for this purpose it is necessary to conduct some adaptations. For example, each «Activity» metaclass must be related to at least one «Product» in our metamodel. The latter metaclass is represented by archetype concept within our proposal, i.e., the result of an activity is the completion of clinical data (which are defined in its archetype, Figure 3).

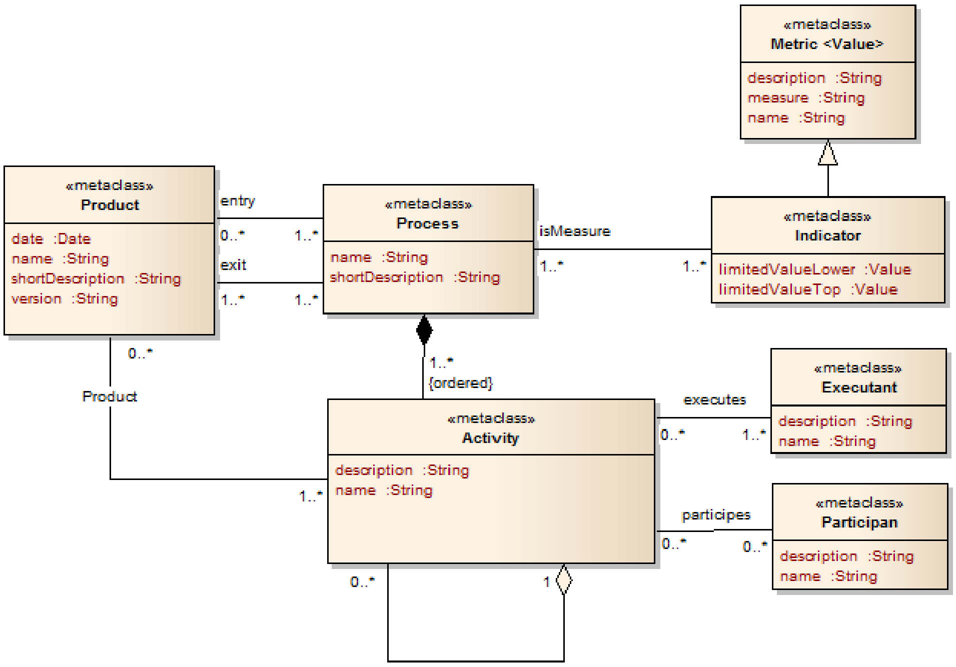


Figure 2. Overview of our process metamodel

On the other hand, our second metamodel allows modelling archetypes (clinical data) with which it is possible to define and express constraints on clinical data and its electronic health record. For this purpose, our proposal takes into account the metamodel defined by the CEN/ISO EN13606 standard, but we improve it with concepts and features defined by the ISO 21090 and ContSys standards. Regarding the ContSys standard, it is important to mention the «Element» metaclass[1] has an attribute which establishes level of compliance according to care concepts defined in the ContSys standard («CONTSYS» enumeration in Figure 3).

Figure 3 shows briefly our archetype metamodel. It does not show properties of each metaclass in order to simplify the representation, but all information is available in CEN/ISO EN13606 standard specification.

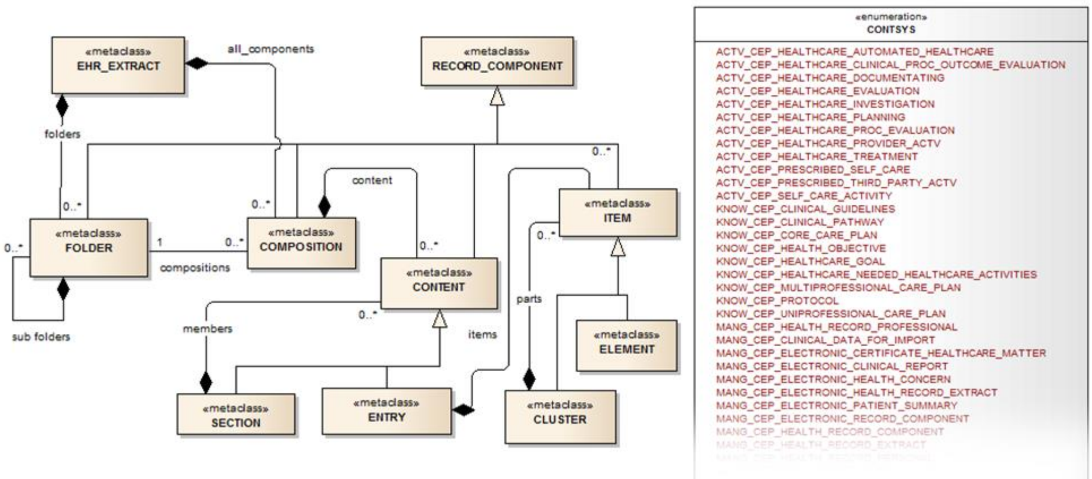


Figure 3. Archetype metamodel

[1] The «Element» metaclass is defined by the CEN/ISO EN13606 standard and it is the leaf node of the Electronic Health Record (EHR) hierarchy, containing a single data value (for instance, systolic blood pressure, heart rate, drug name, symptom, body weight, etc.).

Once presented both metamodels, it is necessary to establish relationships between them in order to support the third step of our methodology and connect the dynamic execution (process) with the static aspect (archetype). For this purpose, we have also established this relationship between our metamodels in order to apply successfully our proposal in real environments.

The **fourth step** of our methodological proposal is a systematic and automatic step and it is supported by a set of transformation rules with which it is possible to generate executable code in order to deploy our models onto software platforms. In this sense, we have defined a systematic procedure which is composed by a set of model-to-text transformation rules. These rules have been specified using the MOFM2T (MOF Model to Text Transformation Language [23]) language and have been adapted to our target architecture: the eHealth architecture (Section 4). For this purpose, we have generated XHTML, SQL, CSS, and Java code from instances of our metamodels (HP and archetype). However, it is important to clarify that our transformation rules can be adapted to any software code. It is just necessary to define the target platform and adapt our systematic transformation protocol.

Finally, **the fifth step** of our methodology is to adapt the generated code to specific features of the specific platform. In this sense, the IT expert team must manually adjust a part of the final executable version in order to add specific aspects of the target platform.

3. Support tool

Previous section has briefly and theoretically presented our methodology to make easier the clinical process management.

However, it is required to offer a tool-based mechanism to support this methodological framework in order to reduce costs and improve the applicability of this framework itself in real environments. In this sense, we have used Enterprise Architect (EA) as our base modeling tool. We have chosen EA because it provides UML2.0 extension mechanisms; incorporates MDE mechanisms or algorithms in order to systematically generate models from other models; and is compatible with UML2.0, among others aspects. Furthermore, another reason led us towards using this tool instead of another one: EA is widely utilized and known by the Andalusian Regional Ministry of Health and Social Welfare.

After choosing the base modeling tool, we have implemented our methodology on it (both our model-based languages and our transformation systematic procedure).

On the one hand, it is necessary to define a concrete syntax to represent each metamodel. In this sense, we have searched for usable representations to resolve the communication problems between healthcare roles and we have chosen UML Profiles. This representation is very common, world-accepted and the most frequently used when metamodels are represented.

- Regarding to HP, our UML profile is based on UML Activity Diagrams (AD) because it's less cognitively complex than other representation such as BPMN (Business Process Modelling Notation) [13].
- Regarding to archetype, we use User Interfaces Design (UID) [14] elements in order to define the archetype UML profile.

Both representations are friendly notation and can be easily used by non-IT (Information Technology) experts who consider and understand HP as activity diagrams and archetypes as UIDs. These UML Profiles have been implemented in EA but it is important to emphasize that any tool that provide UML2.0 extension mechanisms can implement our UML Profile.

Figure 4 shows our UML Profiles through EA's toolboxes. The left toolbox shows the UML profile of clinical processes and the right toolbox represents the UML profile of archetypes.

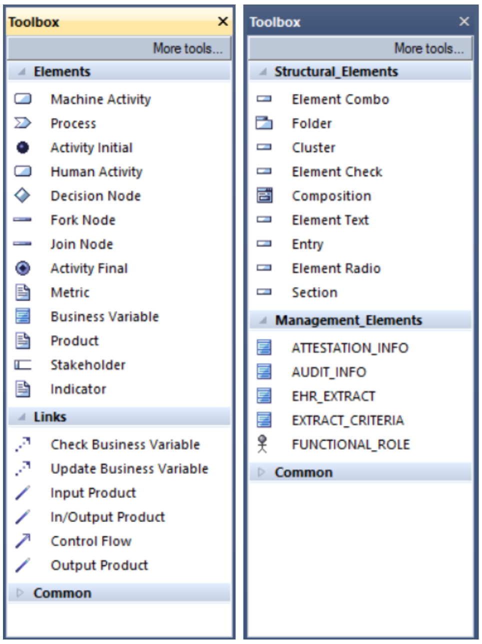


Figure 4. UML Profiles through EA's toolboxes

On the other hand, our tool provides an automatic mechanism which implements the model-to-text transformation rules in order to generate code from process and archetype models. This automatic mechanism is runnable through our EA's plugin. The code generated (HTML, SQL, CSS, and Java) by our tool is adapted to the eHealth platform. We have had to generate this type of code because of the eHealth platform has been design and develop following the MVC (Model/View/Controller) architecture [21].

However, it is important to clarify that our transformation rules can be adapted to any software code.

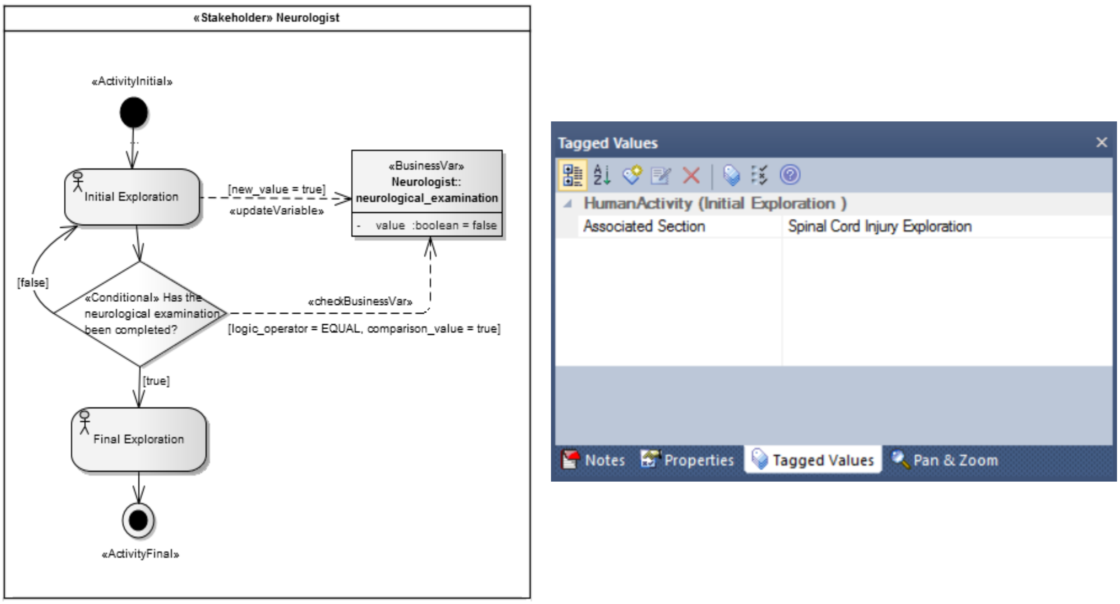
4. Case study

After describing our methodological proposal and presenting our model-based tool, this section describes a case study in which we have successfully applied our proposal. Although our methodological proposal is itself general, it is adapted to our target software environment: the *eHealth platform* of the Andalusian Regional Ministry of Health and Social Welfare. This platform integrates research healthcare processes and has been designed using the MVC (Model/View/Controller) architecture [9] by means of a SOA strategy, offering support to the different researches and innovation projects that have already been performed. In this context, we have had to adapt our methodology to this specific platform.

This case study is included in a project carried out to adapt the Andalusian eHealth platform to SOA-based processes architecture so as to allow better modularity, independence, maintainability and reusability of developed services.

In [15], you can find a video demo of our proposal. However, in this section we have included additional information to make the video more understandable.

We have modelled a common HP (Figure 5A) that describes, in a simplified manner, how doctors must review and assist their patients with spinal cord injury. Moreover it is important to remember that every «Activity» must be related to at least one «Product» in our metamodel. The latter metaclass is represented by archetype concept within this project, i.e., the result of an activity is the completion of clinical data (which are defined in its archetype). The relationship between activity and its product (i.e., its archetype) has been implemented through a tagged value in each human activity. Figure 5B shows how this relationship has been implemented in Enterprise Architect.



(A) **(B)**

Figure 5. Successful case in the eHealth process context

In addition, we have also defined the archetype associated with the previous clinical process. Figure 6 partially shows the archetype for this project[2]. It must be remarked that this archetype is associated with the «Initial Exploration» activity (shown in Figure 5A).

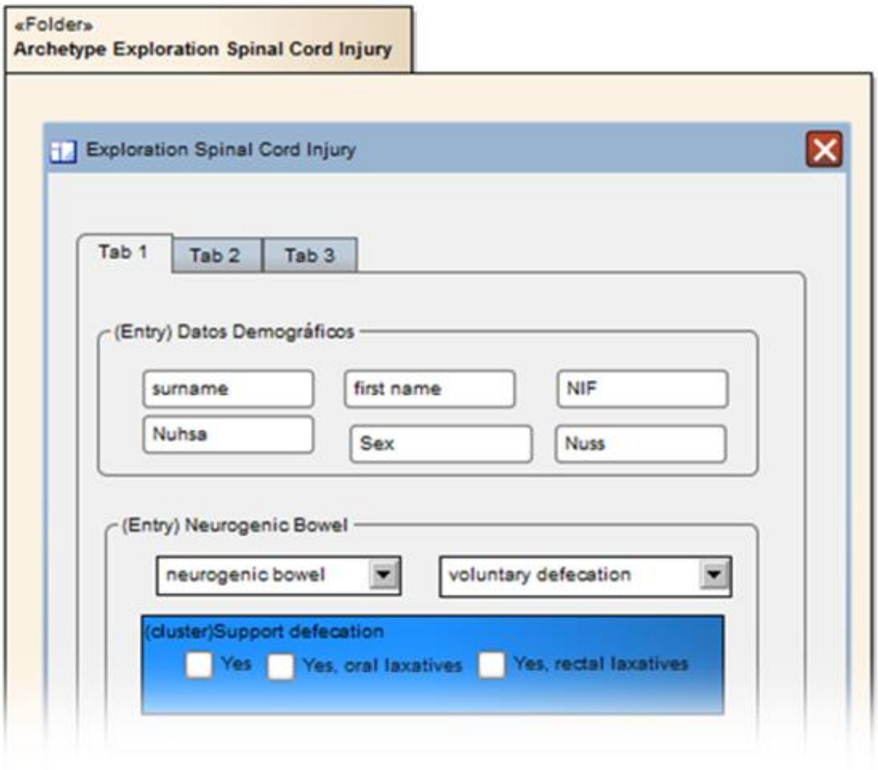


Figure 6. Archetype associated with the examination process of spinal cord injury

[2] We cannot provide a more complex process due to confidentiality constraints.

Finally, after executing our plugin on the clinical process and the archetype, our tool generates MVC files which can be deployed in the eHealth platform. Figure 6 provides a version of the final XHTML prototypes generated. This business logic is being generated semi-automatically and today, it is only available in Spanish. Figure 7 shows information in Spanish because our platform has been developed in Spanish, but we have included some marks within this figure in order to facilitate its understanding. The (A) mark is a tab and represents the first activity of our clinical process (Figure 3). The information related with our archetype is grouped with different XHTML panels with which each the «entry» and «cluster» elements[3] are visually represented to end users (i.e., clinical professionals). For example, the (B), (C) and (D) marks identify the Injury Data, Etiological Data and Spinal injury, respectively.

We have chosen this XHTML representation in order to improve the usability of our proposal when it is used by non-software expertise users (such as doctors). There are studies based on well-established theories (such as Moody's [18] or Goodman's [19]), which have analyzed the cognitive overload of several languages (such as i* [20] or UML [20], among others). One of the conclusions of these studies is that complex modelling languages (i.e. with many elements, many possible interconnections between elements or extensive symbology) increase user's cognitive overload and make the defined model harder to understand. This usability aspect has a major influence on users' efficiency when they use a specific language (for example our language with which users can model their clinical archetypes.

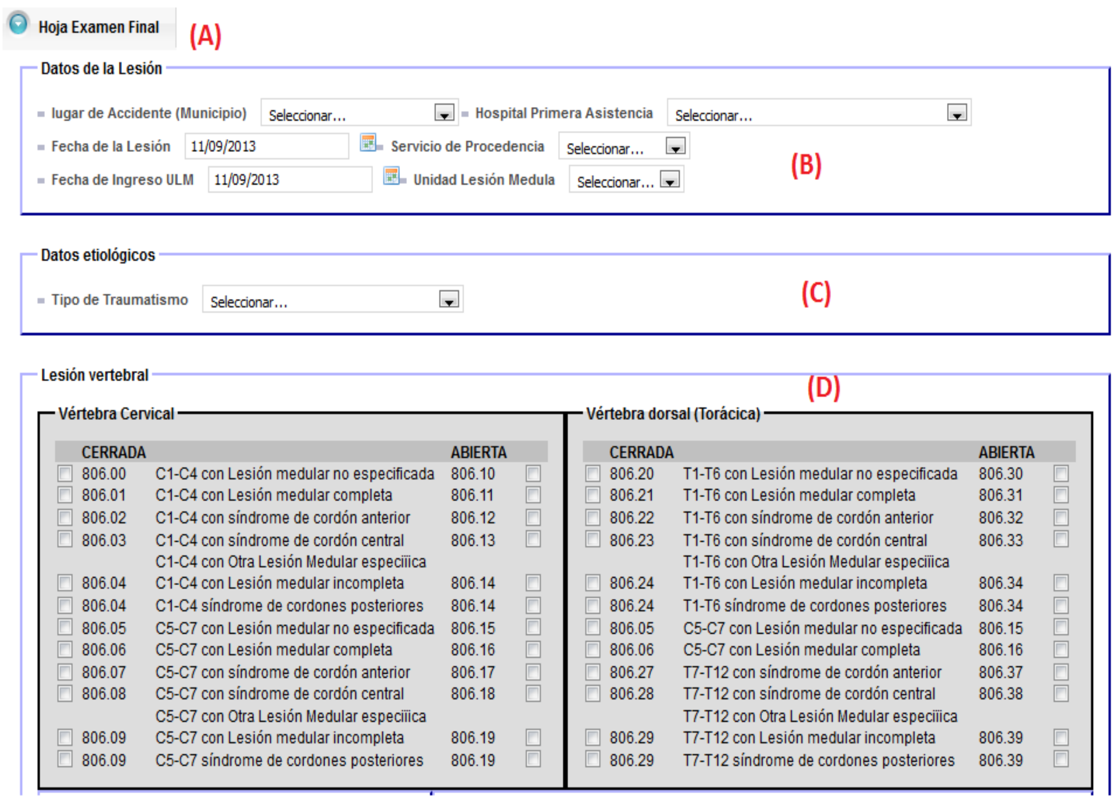


Figure 7. Process visualization on the eHealth platform

[3] The «entry» and «cluster» elements are concepts defined by the CEN/ISO EN13606 standard. On the one hand, «entry» means the information recorded in an Electronic Health Record (EHR) as a result of one clinical action, one observation, one clinical interpretation, or an intention. This is also known as a clinical statement (for instance, a symptom, an observation, one test result, a prescribed drug, an allergy reaction, a diagnosis, a differential diagnosis, a differential white cell count, blood pressure measurement, etc.). On the other hand, «cluster» means of organizing nested multi-part data structures such as time series, and to represent the columns of a table (for instance, audiogram results, electro-encephalogram interpretation, weighted differential diagnoses, etc.).

This project has enabled us to gain valuable feedback for us from the user's view because we were able to define business processes required alongside clinical professionals who successfully and easily interpreted our models. In fact, we were able to reduce development times thanks to our modelling language is user friendly for non-experts. Therefore we could validate our models without spending much time.

5. Conclusions and future works

This paper is contextualized in the healthcare area and presents a demonstration of our model-based and MDE-based tool which incorporates our theoretical methodology to manage clinical processes and clinical data in an effective way. After using our MDE-based solution, it has been substantially possible to reduce development costs without breaching health informatics standards and great satisfaction of end users and with a little time spent on it. In fact, we have made a comparison between code automatically generated and the final code required in the case of module of Spinal Cord Injury scenario and the eHealth platform. As a result we have obtained that MDE-based solution has been able to generate 78.95% of the final code. The remaining 21.05% of the final code was manually implemented.

This work is born after identifying needs of suitable solutions. It guarantees continuous improvement and the quality of results.

In addition, our tool improve communication problems between healthcare professionals and IT because it uses common notations based on standards, either ISO-EN 13606 for archetypes, ISO-EN 13940 for concepts definition, or ISO/IEC TR 24744 for processes definition. In this sense, we also propose a graphical notation focused on prototypes and UML profile to improve communication with final users at clinics.

The way to implement processes from their definition is also solved with model-to-code transformations. Besides, the problem of traceability is answered since transformations help us keep a relation between each concept at HP definition level and each one at the implementation level.

The present study opens a very relevant research line. As future work, we would like to improve our metamodel by incorporating the last recommendations of Integrating the Healthcare Enterprise (IHE) [16] and other relevant standards like ISO 12967 (HISA) [17].

Acknowledgements

This research project has been supported by the “Carlos III” Health Institute within the call Strategic Help in Health (PITeS project, code PI09/90518), RETICS Innovation in Healthcare Technology call (code RD09/0077/00025), by MeGUS Project (TIN2013-46928-C3-3-R) of the Ministerio de Ciencia e Innovación, Spain. We wish to acknowledge the collaboration of the healthcare professionals to the neuro-rehabilitation department to the VRUH for their help in the definition of the spinal cord injury HPSS It has also been supported by the Universia foundation through its student grant for PhD students.

References

1. Weske M. Business Process Management: Concepts, Languages, Architectures; Springer Heidelberg Dordrecht London New York; 2012.
2. ISO Technical Committee 215. ISO-EN 13606: Health Informatics - Electronic Health Record Communication. 2008-2010.
3. ISO/DIS 13940 Health informatics -- System of concepts to support continuity of care.
4. Blobel B, Goossen W, Brochhausen M. Clinical modeling—A critical analysis. International Journal of Medical I, nformatics. 8 3. 2 0 1 4. pp. 57–69.
5. Schmidt DC. Model-Driven Engineering. IEEE Computer, Computer Society, vol. 39, no. 2, pp. 25-31, 2006.

6. ISO/IEC. ISO/IEC TR 24744:2007 Software and systems engineering Lifecycle management Guidelines for process description. International Organization for Standardization, 2007.
7. "Virgen del Rocío" University Hospital. Andalusian Health Service. Available in www.huvr.es. Last accessed 12-2014.
8. Mohagheghi P, Gilani W, Stefanescu A, et al. An empirical study of the state of the practice and acceptance of model-driven engineering in four industrial cases. Empirical Software Engineering, 18(1), 89-116. 2013.
9. James W. Cooper. Java Design Patterns: A Tutorial. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. ISBN: 0-201-48539-7.2000.
10. ISO 21090: Health informatics - Harmonized data types for information interchange. 2011.
11. García-Borgoñon, L., García-García, J. A., Alba, M., Escalona, M. J. "Software Process Management: A Model-Based Approach". Information Systems Development: Building Sustainable Information Systems (Proceedings of the 2012 International Conference on Information Systems Development), Pp 167-178. Editors: Henry Linger, Julie Fisher, Andrew Barnden, Chris Barry, Michael Lang, Christoph Schneider. ISBN: 978-1-4614-7539-2. 2013.
12. Unified Modeling Languages 2.4.1. UML. Available in www.omg.org/spec/UML/2.4.1. Las accessed 12-2014.
13. OMG. BPMN, Business Process Modelling Notation, Version 2.0. Object Management Group, http://www.omg.org/spec/BPMN/2.0/. 2011.
14. Oppermann R. User-interface design. Handbook on information technologies for education and training (pp. 233-248). Springer Berlin Heidelberg. UML. 2002.
15. IWT2´s youtube. URL demo: https://www.youtube.com/watch?v=GvNGyvD_-ak
16. IHE. Integrating the Healthcare Enterprise. Available in www.ihe.net. Last Accessed 12-2013.
17. ISO 12967-2009 Health informatics -- Service architecture (HISA). Libro de
18. Moody, D.L., Heymans, P., Matulevicius, R. Improving the Effectiveness of Visual Representations in Requirements Engineering: An Evaluation of i* Visual Syntax. Proc. of the 17th IEEE International Requirements Engineering Conference, pp. 171–180. 2009.
19. Goodman, N. Languages of Art: An Approach to a Theory of Symbols. Bobbs-Merrill Co. 1968.
20. Moody, D.L., van Hillegersberg, J. Evaluating the visual syntax of UML: an analysis of the cognitive effectiveness of the UML family of diagrams. LNCS, vol. 5452, pp. 16–34. Springer. 2009.
21. James W. Cooper. Java Design Patterns: A Tutorial. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. ISBN:0-201-48539-7.2000.