

A Practical Medical Experience of successfully mixing Model-Driven Paradigm and Business Process Management principles

V. Cid-de-la-Paz¹, L. Morales¹, J.M. Ramos²

¹ IWT2 Group. University of Seville. Spain.

² SOLTEL S.A., Spain

{virginia.cid,leticia.morales }@iwt2.org, juanmanuel.ramos@soltel.es

Keywords: Clinical Decision Support, Model Driven Engineering, Clinical Practices Guidelines.

Abstract: The Model-Driven Paradigm has been successfully used in several different software contexts and there are a lot of literature offering approaches, techniques and tools to guarantee its application in different areas, such as software design, software testing, and so on. But, this paradigm can be also used in other contexts offering very good results. In this paper, we illustrate the power of using models and transformations to make an effective and efficient management of clinical guides in medical environments. The paper shows how using business process management to represent clinical guidelines, principles of Model-Driven paradigm can be successfully used. The paper presents the experiences in the IDE4ICDS, which is framed into the medical context to provide a solution to manage the life cycle of clinical guidelines. This project presents a methodology that allows the management of clinical guidelines to be automated, as well as a software platform to support it. This platform has been validated with health professionals from the Hospital Virgen del Rocio (Seville), obtaining promising results. Nowadays, this platform is being validated by healthcare professionals of Primary Care with patients suffering from Diabetes Mellitus Type 2.

1 INTRODUCTION, CONTEXT AND NEED

BPM (Business Processes Management) could be considered as a management strategy that includes methods, techniques and tools to support the business processes lifecycle, which includes design, enactment, management and analysis of operational BPs [Van-der-Aalst.2002]. BPM aims to reduce costs and improve processes (through a cycle of continuous improvement) in many organizations [PMI.2008]. In fact, some studies, such as [ISO/IEC.ISO 9001.2008], conclude that the reasons for adopting BPM can be grouped into three main needs: (i) understand and assimilate the intrinsic knowledge of processes; (ii) know the employees' performance during the execution processes; and (iii) monitor and measure processes. Controlling these needs improves ROI (Return on Investment) parameter through reducing production costs [Trkman P. 2010] in any kind of organization.

There are also many institutions that promote, by means of their standards and guidelines, the application of BPM as a process-oriented

mechanism to improve productivity, competitiveness, quality and efficiency at organizations [OMG.2011, Martinez-Ruiz T *et al.* 2008, ISO/IEC.ISO/IEC TR24744.2007, Ponce J *et al.*2013]. Such parameters have been followed by a large number of companies in all areas of business.

Healthcare organizations are not an exception and, in fact, the Healthcare Process (HP) management is essential to ensure adequate patient care, as well as facilitate the work of healthcare professionals in an area where it is essential to take of decisions based on the best available biomedical knowledge. This practice is known as Evidence-Based Medicine (EBM) [Tonelli M *et al.*2018] and it is applied within a framework of quality of care, patient safety and efficiency. The management of HP is also closely related to the term of clinical guidelines, which are usually textual and systematic statements of information on HP, clinical records, recommendations and clinical decision rules.

The Clinical Guidelines (CG) themselves aim to improve the quality and safety of patient care, reduce variability in clinical practice and reduce healthcare costs. In this context, in recent years, different research groups in the field of medical

informatics have developed conceptual models to represent executable CG. However, the implementation and maintenance of these conceptual models in software systems are hard, complex and expensive tasks because these systems are usually designed and developed ad hoc what implies hindering interoperability among organizations, the lack of standardization, and increasing the inter-center or even inter-professional variability.

Moreover, at present, there are not methodological frameworks that define how managing comprehensively all phases of the CG lifecycle: from its modeling (including its HP, clinical records, clinical rules, etc.) to its execution, monitoring, evolution, and integration with Healthcare Information Systems (HIS) and Clinical Devices of Patients (CDP, such as glucometer, blood pressure monitor, clinical sensors, etc.). For this reason, designing and developing software systems to CG management are complex task and, in many cases, these systems are developed ad hoc by each health organization that decides to implement a specific CG what implies that the same CG can be developed (and evolved) in different ways in different health organizations.

In this context, it is necessary to research, propose and define mechanisms to ensure the correct and successful execution and management of CG in order to reduce developing costs of software systems and reduce the variability of application of CG in similar medical situations in different patients. However, it is important to mention that although these systems are defined correctly, it is necessary to give solution to another important aspect related to the maintenance of CG because this one evolves frequently. Consequently, proper management of change, maintenance of traceability between the definition and implementation of CG, as well as the achievement of effective continuous improvement, become fundamental and key tasks within health organizations.

This paper describes a practical experience in a real R&D project which aims to propose a technological solution to solve the previous need in health organizations. This project (as well as its support software platform) is named IDE₄ICDS (Integrated Developing Environment for Improving Clinical Decision Support based on Clinical Guidelines) and it is subsidized by the Ministry of Economy and Competitiveness and co-financed with FEDER funds, in the call Challenges-Collaboration of the State Program of Research, Development and Innovation Oriented to the Challenges of Society, within the framework of the State Plan for Scientific Research and Technique and Innovation 2013-2016.

The Consortium of public and private entities that is carrying out the project are: the IWT2 Group

of the University of Seville, Soltel IT Software SLU, Serviguide Consultoría S.L. and the GIT Group of the FISEVI Foundation.

IDE₄ICDS is based on the Model-Driven Engineering (MDE) [Schmidt D.C.2006] paradigm and defines a methodological framework to comprehensively and collaboratively manage CG, as well as systematic mechanisms for deploying and maintaining these CG on a web platform. Moreover, our experience in transferring knowledge to companies confirms that naive MDE-based solutions are more likely to succeed because they enable designing and implementing transformation rules [Escalona, M. J *et al.* 2007, García-García J.A., *et al.*2015, Gutierrez, J.J, *et al.*2015, Dominguez, FJ *et al.*2010, Dominguez, FJ *et al.*2014, García-García J.A *et al.*2014].

The features of IDE₄ICDS are: (i) centralized, i.e., a single nucleus of information is stored and traceability maintained between all the components of a clinical guide (processes, simple elements and decision rules); (ii) integral, i.e., a single platform (IDE₄ICDS) provides modules to define, execute, monitor and interoperate the CG with HIS and CDP; and (iii) collaborate, i.e., if a health professional considers it necessary to improve or evolve a CG (based on his or her experience), he or she can do it intuitively and friendly, and that modification could be used by another health organization. In addition, it is important to mention that this project will be tested and validated in a real scenario of patients with Type 2 Diabetes Mellitus¹.

After this introduction, this paper is structured as follows. Section 2 describes our proposal to clinical guidelines management. Section 3 describes our technological solution. Finally, Section 4 states conclusions and introduces future lines of research.

¹ Andalusian Regional Ministry of Health. [Integrated Welfare Process Diabetes Mellitus](#). Last access 2018.

2 THEORETICAL FOUNDATIONS OF IDE4ICDS BASED ON AN IMPROVEMENT CONTINUOUS LIFECYCLE

From a general point of view, process management could be considered a management strategy with a clear multidisciplinary nature, as it can be applied to different contexts (e.g., healthcare domain) and can be used by different user profiles [Hill J.B, *et al.*2017]. This situation has conditioned the appearance of different views, definitions and perspectives of management lifecycles as well as their continuous improvement [Van-der-Aalst.2004], which define a management model for continuous business implementations and incremental problem solving. Although clinical guidelines have more feature than simple process, the Clinical Guideline Management (CGM) is similar because both ones need to be modelled, executed, orchestrated, measured, improved, etc.

However, CGM has several particularities which have to be properly supported by decision-support software systems. As mentioned in previous section, these particularities are related to improve the maintenance and evolution of CG, streamline the day of health professionals and reduce the variability in the clinical practice and sanitary cost.

For this purpose, the project defines a model-driven theoretical framework to support a continuous improvement lifecycle of CG based on four phases which are:

1. **Modeling Phase.** In this phase, healthcare professionals can model his/her CG in a structured manner (i.e., identifying roles, activities of the healthcare process, clinical rules, clinical recommendations, for instance). For this purpose, we have carried out different studies of international clinical guidelines to extract and analyse its textual structure. Once analysed health documents, we have defined a simple, flexible and highly-semantic metamodel (which takes the form of a MOF-compliant metamodel) to model any aspect of CG. MOF (Meta-Object Facility) is a set of standard interfaces that can be used to define and manipulate a group of interoperable meta-models and the corresponding models. We offer a flexible language to model CG with two main goals: (i) facilitate the application of MDE-based mechanisms and extensible of our

metamodel in future; and (ii) reduce users' cognitive overload when they are utilizing our modeling language. From a MDE perspective, this simplicity helps us to successfully apply our solution to health service and open new research lines related to testing and simulating health processes. This simplicity is not seen as a drawback since our proposal has extension mechanisms.

Due to the complexity of CG modelling metamodel, below we are going to briefly describe show the most important metaclasses. However, Figure 1 shows the general metamodel of CG. This metamodel is related to 2 secondary metamodels: «ClinicalProcess» and «ClinicalRule». These metamodel are not explained in detail but can be consulted in [García-García *et al.* 2015; García-García *et al.* 2018].

Before going further, it is worth clarifying that the syntax used is not enough to semantically define our metamodel. In fact, we use OCL (Object Constraint Language) [28] to add formal semantic constraints that validate process models.

The metaclass «ClinicalPracticeGuideline» is the epicenter of the metamodel and is the element around which the rest of the elements of the metamodel are orbiting. With this metaclass it is possible to represent any CG.

To describe the necessary actions since a person, with a certain pathology, requests assistance until it ends, each CG is composed of a clinical process «ClinicalProcess». To model these actions, we have activities (human, automatic and complex) and gateways. In human activities, the health professional registers patient data. For it, we associate the metaclass «Variable».

In addition, given that one of the objectives of this project is to monitorize the CG, with the metaclass «Indicator» we can associate indicators to activities and processes. These indicators, when the GC is running, will record the values and will be displayed in the monitoring module.

To help in making clinical decisions in activities or to choice with path of actions the patient should follow, we associated the metaclass «ClinicalRule» to activity and gateway. These rules are formed by clauses, constituted in turn by the variable to be evaluated, a mathematical operator and the value with which to compare. Logical

operators are used to relate one clause to another.

2. Execution and Orchestration Phase.

Nowadays, this phase could be considered as critical and essential task because health organizations are being driven by the need to extensively and continually automate, evolve and maintenance their CG. For this purpose, CG have to include execution information (i.e., such as execution parameters for the communication and integration with external systems, for instance). This information are essentials to deploy and execute CG models into execution engines, such as process engines or BPMS (Business Process Management Suite [Meidan A, et al.2017]), rule decision engines, etc.

IDE4ICDS provides MDE mechanism to solve this situation. This mechanism allows systematically generating the executable version of CG from its model (previous phase). It is based on *model-to-text* (M2T) transformation rules using MOFM2T [OMG.MOF.2017]. This transformation protocol will not be explained here, since they are out of the scope of this paper and it would become too extensive, but it is possible to find further information of application in other context into [García-García JA, et al.2017].

Anyway, we have been able to generate executable code (based on BPMN-XML and Java code) from the definition model of the GC. On the one hand, BPMN-XML code is generated because most BPMSs support this standard format [Meidan A, et al.2017] and it is used but the process engine selected in our project (see Section 3). On the other hand, we generate Java code to execute each clinical decision rule (which is modelled in previous phase) in the decision rule engine selected in our project (see Section 3).

Although these kinds of code are related to our design and technological solution, it is important to mention that our MDE-based framework is independent of the platform.

3. Monitoring Phase.

Once CG and its healthcare processes are modelled and executed, it is time to evaluate its effectiveness. This evaluation provides a granular view of the overall productivity of each CG and it is based on the definition of key performance indicators.

In this case, we have included two mechanisms in order to support this phase in the IDE₄ICDS platform. Firstly, our CG modeling metamodel includes concepts (such as metric and indicator) that help the healthcare managers to measure each CG. Indicators are defined and

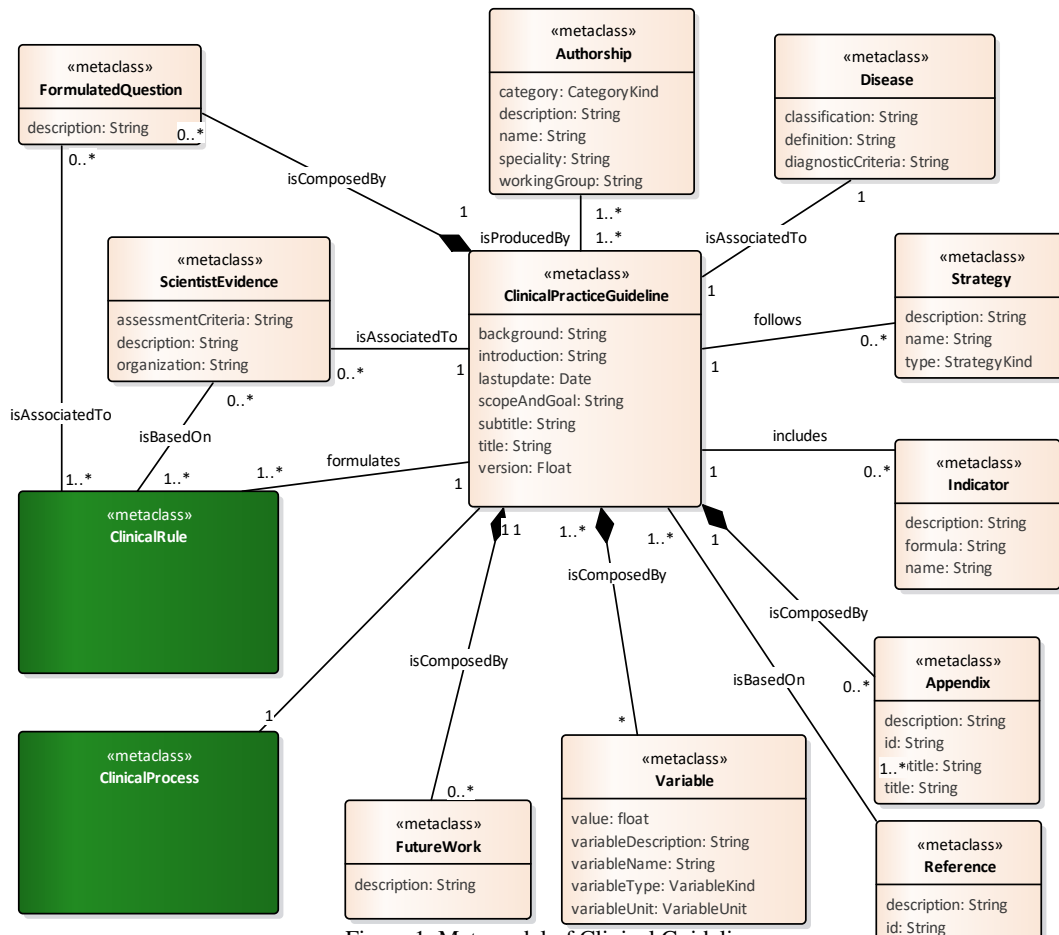


Figure 1. Metamodel of Clinical Guideline

configured during the modeling phase. Secondly, and once modelled CG and indicators, these models are systematically transformed to in executable code using M2T transformation rules. These executable code is composed of: (i) SQL (Structured Query Language) scripts, which update the measurement database of IDE₄ICDS; and (ii) code scripts, which connect each model of the CG (i.e., definition model, execution and orchestration model, indicator model, etc.) between itself, and calculate each defined indicator.

4. **Continuous Improvement Phase.** This phase aims to achieve higher quality, efficiency, effectiveness and performance levels during CG execution what imply to improve the patient assistant. This improvement of CG could emerge from two situations. The first one could success after evaluating CG performance (through assessment indicators and metrics), i.e., a healthcare organization could start an internal improvement process to improve its services to patients and increase or optimize its resources. The second situation could append when healthcare professionals identify any improvement in the definition model of the CG. Anyway, after appending any these situations, organization could iterate over our CG lifecycle as many times as necessary in order to achieve its goals.

3 THEORETICAL FOUNDATIONS OF IDE₄ICDS BASED ON AN IMPROVEMENT CONTINUOUS LIFECYCLE

Previous section has briefly presented our theoretical framework to make easier the CG management. However, it is required to offer a tool-based mechanism to support this framework in order to reduce costs and improve its applicability in real healthcare environments during the definition, design, implementation and validation of clinical guidelines. The IDE₄ICDS platform has been designed and developed to achieve the previous goal under a service-oriented architecture with a user-centered design. For this purpose, our platform

provides five functional modules (Figure. 2) as follows.

The first one is the **Definition & Traceability Module (M1)**, which provides three graphic editors to model each aspect of a GC (i.e., general information, *healthcare processes*, *clinical information records and clinical decision rules*). *These editors* has been implemented as plugins on Enterprise Architect (EA)². These plugins have different functionalities, such as, UML Profiles to friendly instance our GC metamodel (Figure. 1), plugin to guarantee well-defined models verifying each OCL constraint of metamodel, plugin to automatically execute each transformation rule, etc. In addition, it is important to emphasize that these models are stored in a central repository of models (supported on a MySQL database).

The second module is the **Execution Module (M2)**, where the executable version of the CG is systematically deployed and, later, executed by healthcare professionals. Once modelled a GC, the deployment method of this one has to be carried out in two steps. Firstly, process engineer applies transformation rules from M1 to automatically obtain BPMN-XML and Java code which are used

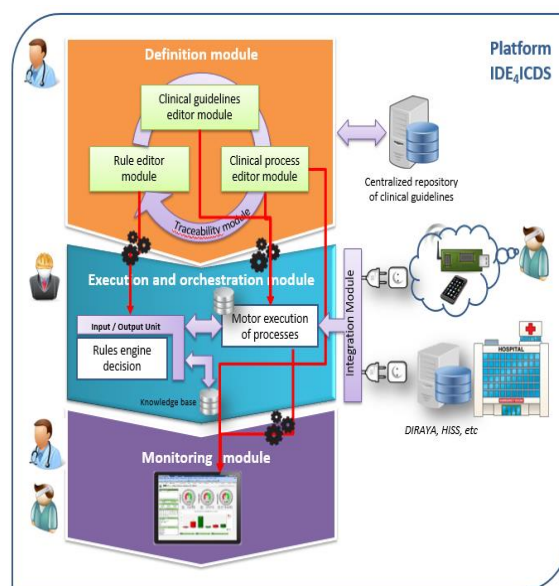


Figure 2. Conceptual model of the IDE₄ICDS platform

² SparxSystems. Enterprise Architect. [Website](#). Last access 2018.

by BonitaOS³ (process engine) and Drools⁴ (decision rule engine), respectively. These tools were chosen according to the requirements of the project and taking into account the characterization scheme proposed in [Meidan A, *et al.*2017], which was very useful. However, it is important to mention that our MDE-based framework is independent of the platform. Therefore, other engines could be chosen within great efforts. Secondly, process engineer has to configure and compile these codes manually. This configuration includes to set execution parameters with HIS and *CDP*. The configuration of this information cannot be automated because it depends on the specific system and chosen device.

Moreover, the fourth module is the **Monitoring Module (M3)**. This module has been specially developed for the project using web technology (cakePHP, HTML5, CSS3, jQuery, etc.). Once modeled a GC using the M1 module, the process engineer can automatically execute the set of M2T transformations to generate cakePHP code from definition models of the GC. This cakePHP code allows calculating each indicator associated with the healthcare process (which is defined in the clinical guideline). For this purpose, the M3 module has to be communicated to the M2 module in order to calculate indicators during the execution of the GC (such as number of patients attended, percentage of emergencies, average time of patient care, maximum time of patient care, etc.). Once calculated each indicator, this module provides scorecards, timelines with the evolution of each indicator, and alarms/notifications, among other functions.

As mentioned above, M3 and M2 have to be communicated. For this purpose, we have designed the **Integration Module (M4)**, which is developed as an enterprise services bus based on OpenESB⁵. M4 allows internally connecting each module with

another one, as well as externally communicating the M2 module with HIS and *CDP*.

Finally, the last module is the **Traceability Module (M5)**. An important aspect when MDE is used is to ensure traceability among generated models. This is essential in the context of *IDE4ICDS* because it allows identifying each GC in a unique way in the platform and its modules. In addition, traceability allows enhancement points such as version management of a GC.

4 CONCLUSIONS AND FUTURE WORKS

Today's world economic situation is ruled by issues such as reducing cost, improving quality, maximizing profit and improving and optimizing processes at any kind of organization. In this context, BPM have been confirmed as an essential and successful strategy. However, over last years, research community have started to combine the process management and MDE within controlled contexts as software testing processes, requirement processes, etc.

However, the application of MDE in health contexts is a work and research line that has aroused the interest of research teams and organizations interested in transferring research results to real environments. In addition, the application of MDE in health environments is also a new research line, little treated over last years and innovative in terms of potential results that allows its application. In this context, this paper has presented the *IDE4ICDS* project which proposes a MDE-based solution to solve two main goals: (i) improving the application and management of CG in real healthcare environments; and (ii) reducing variability in clinical practice during the application of a specific CG, as well as reducing healthcare costs. At present, this project will be tested and validated in a real scenario of patients with Diabetes Mellitus, but we plan to extend and apply our solution to other medical pathologies as future works.

ACKNOWLEDGEMENTS

This research has been partially supported by POLOLAS project (TIN2016-76956-C3-2-R) and *IDE4ICDS (RTC-2016-5824-1)* of the Spanish Ministry of Economy and Competitiveness, and by *the VI PPIT-US of the University of Seville (Spain)*.

³ BonitaSoft. [Website](#). Last access 2018.

⁴ Drools. [Website](#). Last access 2018.

⁵ OpenESB. [Website](#). Last access 2018.

REFERENCES

- Domínguez-Mayo, F. J., Escalona, M. J., & Mejías, M. (2010, July). QuEF (quality evaluation framework) for model-driven web methodologies. In *International Conference on Web Engineering*(pp. 571-575). Springer, Berlin, Heidelberg.
- Domínguez-Mayo, F. J., Escalona, M. J., Mejías, M., Ross, M., & Staples, G. (2014). Towards a homogeneous characterization of the model-driven web development methodologies. *Journal of web engineering*, 13(1-2), 129-159.
- Escalona, M. J., Gutierrez, J. J., Villadiego, D., León, A., & Torres, J. (2007). Practical experiences in web engineering. In *Advances in Information Systems Development* (pp. 421-433). Springer, Boston, MA.
- García-García, J. A., Meidan, A., Carreño, A. V., & Risoto, M. M. (2017, October). A Model-Driven Proposal to Execute and Orchestrate Processes: PLM4BS. In *International Conference on Software Process Improvement and Capability Determination*(pp. 211-225). Springer, Cham.
- García García, J. A., Escalona, M. J., Martínez-García, A., Parra, C., & Wojdyński, T. (2015). Clinical Process Management: A model-driven & tool-based proposal. *Information Systems Development: Transforming Healthcare through Information Systems*. ISBN: 978-962-442-393-8. 2015.
- García-García J.A., García-Borgoñón L., Escalona MJ, Mejías M. A model-based solution for processes modeling in practice environments: PLM4BS. *Journal of Software-Evolution and Process*. pp.1-23. <https://doi.org/10.1002/smr.1982> 2018.
- García-García, J. A., Escalona, M. J., Domínguez-Mayo, F. J., & Salido, A. (2014). NDT-Suite: A methodological tool solution in the Model-Driven Engineering Paradigm. *Journal of Software Engineering and Applications*, 7(04), 206.
- Gutiérrez, J. J., Escalona, M. J., & Mejías, M. A Model-Driven approach for functional test case generation. *Journal of Systems and Software*, 109, 214-228. 2015.
- Hill J.B., et al. *Cool Vendors in Business Process Management*, Gartner Research, 2007.
- ISO/IEC.ISO 9001:2008. *Quality management systems, Requirements*. International Organization for Standardization, 2008.
- ISO/IEC.ISO/IEC TR24744:2007 *Software and systems engineering lifecycle management guidelines for process description*. International Organization for Standardization, 2007.
- Martínez-Ruiz T., et al. Towards a SPEM 2.0 extension define process lines variability mechanisms. *Software engineering research, management & applications*, 115–130, 2008.
- Meidan, A., García-García, J. A., Escalona, M. J., & Ramos, I. (2017). A survey on business processes management suites. *Computer Standards & Interfaces*, 51, 71-86.
- OMG. BPMN, *Business Process Modelling Notation, Version 2.0*. 2011.
- OMG.MOF Model to Text Transformation Language (MOFM2T). Website. 2017.
- PMI. *A Guide to Project Management Body of Knowledge*. ISBN: 978-1933890517. 2008.
- Ponce, J., Borgoñón, L. G., García, J. G., Escalona, M. J., Domínguez-Mayo, F. J., Alba, M., & Aragon, G. (2013). A model-driven approach for business process management. *Covenant Journal of Informatics and Communication Technology*, 2(1).
- Schmidt D.C. *Model-Driven Engineering*. *IEEE Computer*, 39, 2, 25-31, 2006.
- Tonelli M., et al. *The role of experience in an evidence-based practice*. *The Medical Roundtable General Medicine Edition*, 2018.
- Trkman, P. The critical success factors of business process management. *International Journal of Information Management*,30(2),125-134. 2010.
- Van-der-Aalst WMP. *Business process management: a personal view*. *Business Process Management Journal*, vol.10 (2), 2004.
- Van-der-Aalst WMP. *Making Work Flow: On the Application of Petri Nets to Business Process Management*. LNCS, vol. 2360, Application and Theory Petri Nets, pp 1-22. 2002.