Systematic Development of ERP Modules using a Model-Driven Strategy Focusing on the Users

M. A. Olivero[1,2][a], L. Morales-Trujillo[1][b], F. J. Domínguez-Mayo[3][c] and M. Mejías[3][d]
[1]*Research Group IWT2, University of Seville, Spain*
[2]*Istituto di Scienza e Tecnologie dellInformazione, Consiglio Nazionale della Ricerche, Italy*
[3]*Department Languages and Computer Systems, University of Seville, Spain*

Keywords: Design Sprint, ERP, NDT, Odoo.

Abstract: ERP systems are composed of different functional modules on which each one addresses a different business area. Developments on these modules are managed independently on each one, which allow to handle and address the management of many related information requirements. In this context the startup G7Innovation works for its product iMEDEA. In this study we have combined two methodologies, Design Sprint and NDT 4.0; and systematized the development of ERP system modules. This combination allows you to use Design Sprint to generate and validate prototyping, and NDT 4.0 to do the study, analysis, and design of the software to be developed. In addition, according to the specifications defined in NDT, the code generation of the ERP module can be automated. This proposal has been validated in a case study in collaboration with the startup G7Innovation, where we have applied both methodologies on Odoo, an open source ERP system based in Python language. Thanks to the use of these two methodologies we have produced a module related to the needs of the clinic by reducing costs, times and human failures.

1 INTRODUCTION

The iMEDEA project is a solution developed by G7Innovation (G7Innovation, 2019). This project is based on the Odoo ERP (Odoo, 2019) and its goal is to facilitate the management of human reproduction units. It is composed of different modules that can be used independently, which offer a better support to business customers, giving fast response times to their problems and with efficient information management (Delgado and Marín, 2000). All these features allow timely decision making and lower operating costs.

Without a clear guidance, modules are not produced according to client expectation, and delays are introduced when modifications are needed after requirements has been defined.

With the aim of reducing changes in later stages of the project and to produce modules according to client expectation, this work presents a methodological proposal that encompasses two methodologies, Design Sprint (Banfield et al., 2015), (Knapp, 2018) and NDT (Escalona et al., 2004), (Navigational Development Techniques). These already existing methodologies focus on the early validation through prototyping and the code generation respectively. This reduce the learning curve needed for simple developments, increase productivity, and reduce the need of changes in the requirements.

In this work we focus on the Odoo modules, being the main product developed by the startup G7Innovation. We have validated this proposal by generating a new module with this company in a joint work.

The rest of the article is organized as follows: The R&D project is defined in section 2, the used tools in section 3 and the research ends with the conclusions in section 4.

[a] https://orcid.org/0000-0002-6627-3699
[b] https://orcid.org/0000-0001-9554-1173
[c] https://orcid.org/0000-0003-3502-8858
[d] https://orcid.org/0000-0002-0353-6391

Olivero, M., Morales-Trujillo, L., Domínguez-Mayo, F. and Mejías, M.
Systematic Development of ERP Modules using a Model-Driven Strategy Focusing on the Users.
DOI: 10.5220/0008516204890492
In *Proceedings of the 15th International Conference on Web Information Systems and Technologies (WEBIST 2019)*, pages 489-492
ISBN: 978-989-758-386-5

2 R&D PROJECT

This guide is differentiated in three stages: (1) *Software Product Discovery*, where we apply Design Sprint to directly know the real needs of users; (2) *Development*, where the project is technically specified, we build and test the system in an unattended way by using NDT and model transformations; and (3) *Operation*, for the deployment and maintenance Figure.1 summarizes this process.

The *software product discovery* stage aims to know what are the needs and motivations that need a solution. To this end, Design Sprint methodology is applied. This methodology is structured in a set of milestones that are scheduled to be completed in five days. As a summary of this methodology, different solution proposals are put forward with the aim of obtaining different approaches that help in decision-making. Final decision of how the solution shall be rely on the information that has been obtained from each alternative. Once the client agrees on the solution to be achieved, the prototype is sketched. The prototype must be realistic enough to finally be able to validate the solution proposal with the user. The use of prototypes to validate the project allows to dis- card the alternatives at low cost.

Development of the product based on the prototype is divided into three milestones: *technical specification, construction and verification and validation*. The *technical specification* defines the structure and behavior of our system using UML (OMG, 2015) models and IFML (Brambilla et al., 2014) models. Knowing the result that we want to obtain, thanks to the prototypes, and once defined the technical architecture, the construction stage begins.

During *construction*, we make use of model to text (M2T) operations, which allow us to obtain code from design models. These transformations generate code from the data model of our system, taking the diagram of classes of the system from NDT as the source model. Similarly, function codes can be obtained through the functional requirements and use cases model offered by NDT. The generation of automated code increases productivity and reduces the possibility of human failure.

Verifying and validating the quality of the project is the last step before launching the product into production. During this stage, functional and acceptance tests must be carried out, among others. To reach a stable testing environment, another type of transformation is used, the Model to model (M2M) transformations offered by NDT. Using the CASE tool of NDT, allow you to generate the documentation of the test plan.

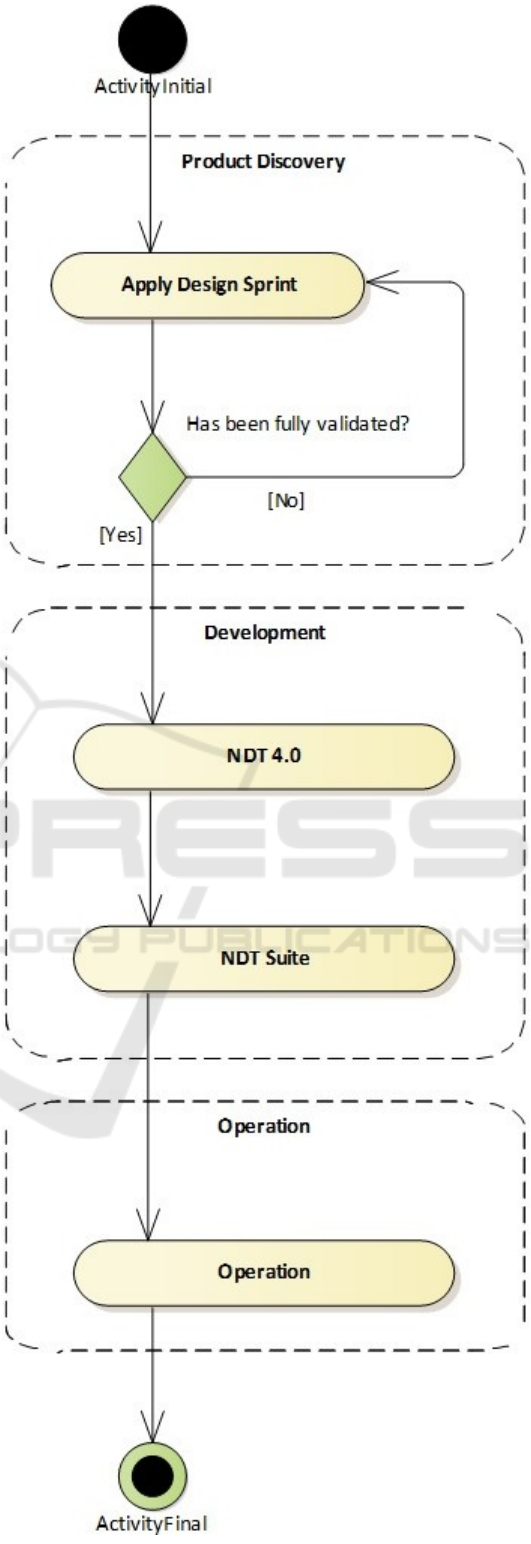

Figure 1: R&D Project sequence.

The proposal concludes with the *operations* phase. This phase is responsible for ensuring the proper

working of the system and its technological infrastructure in production environments when the application is deployed (Hertvik, 2014).

3 USED TECHNOLOGIES

In the discovery phase of the product, the "PowerPoint" tool has been used to develop and validate prototypes. Although there are numerous alternatives, to quickly prototype, we have chosen this tool for its great flexibility. Using the pattern mode of this tool it is possible to create templates of the interface of the software to be developed, which can be later completed by adding the content that we want to validate with clients or customers. In addition, this tool allows you to add links between objects so that you can simulate navigation among the slides by using items that simulates the buttons of the proposed final system. This tool produced very precise results that allowed us to make the designed prototypes quite realistic, that were used to validate the solution proposal with the users in some interviews. Given the fact that we know about the Odoo interface, and the technological limitations of the features that Odoo offers, we have generated realistic prototypes at low cost. This allowed us to easily discard some alternatives that did not meet user or customer expectations. This process could be also be done with the Odoo Studio tool, a tool that facilitates the generation of modules with a graphical interface like Drag Drop. Despite it is fully integrated with Odoo and allow autogenerating the module, we did not contemplate this option because of the difficulties it could introduce when reusing the generated code. For G7Innovation it is not enough having a single module, but also be able integrate it with third parties APIs which to the best of our knowledge it is not feasible with generated module with Odoo Studio.

Once the product is defined, the phase of a formal definition of the requirements. The technical specification, within the development phase has been carried out using NDT-Suite (García-García et al., 2014). NDT-Suite is a set of tools that facilitates the use of the NDT methodology, which is integrated on top of the Enterprise Architect tool (Architect, 2019). With this tool, and starting from the agreed prototype, we have been able to define the system class diagrams, actors, use cases with their interaction requirements and some non-functional requirements.

Thanks to the ability to perform M2M (OMG, 2008a) and M2T (OMG, 2008b) transformations offered by NDT-Suite we automatically obtained some of these requirements, Python code that was used as the basis for the construction phase and the documentation of the test plan for verification and validation.

The code self-generated by the M2T transformations has been used as the basis for the construction. However, it is still necessary to adapt some code configuration parameters in Odoo to reach an integration with the system. Nevertheless, given the fact that a great proportion of the code has been generated automatically the productivity is increased and the possibility of human failures has been reduced.

The verification and validation of the system that is generated is carried out using the Katalon tool, by executing the system test plan that generates NDT from the defined ones. It is important to note that the integration tests of the autogenerated modules with the existing ones have not been possible to be self-generated with the use of these methodologies so far, despite the importance these kinds of modular developments.

Once G7Innovation delivered the validated and verified system to the responsible of the iMEDEA project, the informatics team of the clinic was the responsible for carrying out the phase of operations.

Each one of the described phases is highly independent from the others. This allow having work teams that can be working in parallel.

4 CONCLUSIONS

The proposal of this methodology arises as a need to facilitate communication with customers, and to allow a development that is user-friendly without implying an increase in the costs.

After applying these methodologies, we have been able to generate a product closer to the user and client expectations, automating its construction which reduce human failures and increases productivity. The prototypes defined using the Design Sprint methodology were very realistic as we knew about the technology features on which we wanted to develop the final product and its technological limitations.

The use of these methodologies reduced the need of code modifications in advanced project stages because of the validation with the customer's needs during the interviews on which they could test the prototype. Also, human errors are reduced since most of the code was automatically generated. However, the automated generation requires adapting the code and execute more intensive integration tests to verify the compatibility of the generated code with the already deployed modules.

As result, using this combination of methodologies we produced a module in collaboration with the startup G7Innovation.

Finally, we plan to apply our proposal in other case studies of different business areas and scientific areas in collaboration with the research team of this startup. In particular, our proposal can be enriched from a joint work in areas on which the researchers of G7Innovation RD team has been working previously as: (i) the design and development of technological solutions in the area of requirements engineering (García-García et al., 2012), (Escalona et al., 2013b), (ii) the Software Process Management (Garcia-Garcia et al., 2017), (Meidan et al., 2018), (iii) the PLM systems (Product Lifecycle Management) (Escalona et al., 2013a) and (iv) the CPM (Clinical Process Management) (García García et al., 2015).

ACKNOWLEDGEMENTS

This research has been partially supported by the POLOLAS project (code TIN2016-76956-C3-2-R) of the Spanish Ministry of Science and Innovation.

REFERENCES

Architect., E. (2019). Tool. Available at: https://sparxsystems.com/products/ea/. Last accessed: July 2019.

Banfield, R., Lombardo, C. T., and Wax, T. (2015). *Design sprint: A practical guidebook for building great digital products*. " O'Reilly Media, Inc.".

Brambilla, M., Fraternali, P., et al. (2014). The interaction flow modeling language (ifml).

Delgado, J. and Marín, F. (2000). Evolución en los sistemas de gestión empresarial. del mrp al erp. *Economía industrial*, 331(1):51–58.

Escalona, M., Mejías, M., and Torres, J. (2004). Developing systems with ndt & ndt-tool. In *13th International conference on information systems development: methods and tools, theory and practice, Vilna, Lithuania*, pages 149–59.

Escalona, M. J., García-García, J. A., Mas, F., Oliva, M., and Del Valle, C. (2013a). Applying model-driven paradigm: Calipsoneo experience. In *CAiSE Indus- trial Track*, pages 25–32. Citeseer.

Escalona, M. J., Urbieta, M., Rossi, G., Garcia-Garcia, J., and Luna, E. R. (2013b). Detecting web requirements conflicts and inconsistencies under a model- based perspective. *Journal of Systems and Software*, 86(12):3024–3038.

G7Innovation (2019). Technology-based company, based in Seville (Spain) and with a high level of innovation in the field of development of ICT solutions applied to clinical and health management. Available at: http://g7innovation.com/. Last accessed: July 2019.

Garcia-Garcia, J., Enríquez, J. G., Garcia-Borgoñon, L., Arévalo, C., and Morillo, E. (2017). A mde-based framework to improve the process management: the empower project. In *2017 IEEE 15th International Conference on Industrial Informatics (INDIN)*, pages 553–558. IEEE.

García-García, J., Escalona, M., Ravel, E., Rossi, G., and Urbieta, M. (2012). Ndt-merge: a future tool for conciliating software requirements in mde environments. In *Proceedings of the 14th International Conference on Information Integration and Web-based Applications & Services*, pages 177–186. ACM.

García-García, J. A., Escalona, M. J., Domínguez-Mayo, F. J., and Salido, A. (2014). Ndt-suite: a method- ological tool solution in the model-driven engineering paradigm. *Journal of Software Engineering and Ap- plications*, 7(04):206.

García García, J. A., Escalona, M. J., Martínez-García, A., Parra, C., and Wojdyński, T. (2015). Clinical process management: a model-driven & tool-based proposal.

Hertvik, J. (2014). What Does IT Operations Management Do? (ITOps)). Joe Hertvik: Tech Machinist. Available at: https://joehertvik.com/. Last accessed: July 2019.

Knapp J., Zeratsky J., K. B. (2018). *Sprint: El método para resolver problemas y testar nuevas ideas en solo 5 das*. España: Conecta.

Meidan, A., García-García, J. A., Ramos, I., and Escalona, M. J. (2018). Measuring software process: A systematic mapping study. *ACM Computing Surveys (CSUR)*, 51(3):58.

Odoo (2019). Management software. Available at: https://www.odoo.com/. Last accessed: July 2019.

OMG (2008a). Documents Associated with Meta Object Facility (MOF). Available at: https://www.omg.org/mof/. Last accessed: July 2019.

OMG (2008b). MOF Model to Text Transformation Language (MOFM2T). Available at: https://www.omg.org/spec/MOFM2T/About- MOFM2T/. Last accessed: July 2019.

OMG (2015). OMG Unified Modeling Language, v2.5. Available at: https://www.omg.org/spec/UML/. Last accessed: July 2019.