Risk refinement in the deployment process of software systems: a case study

Felipe Ortiz[1], Marisa Panizzi[1,2] Rodolfo Bertone[3]

[1] Master's Program in Information Systems Engineering. Graduate School. Universidad Tecnológica Nacional. Regional Buenos Aires.
Castro Barros 91. (C1178AAA). CABA. Argentina.
[2] Department of Information Systems Engineering. Universidad Tecnológica Nacional. Facultad Regional Buenos Aires.
Medrano 951 (C1179AAQ), C.A.B.A, Argentina.
[3] Instituto de Investigación en Informática [Information Systems Research Institute]- III-LIDI. School of Information Systems. Universidad Nacional de La Plata (UNLP) 50 y 120 – La Plata, Argentina.
ortizfd@gmail.com; marisapanizzi@outlook.com; pbertone@lidi.unlp.edu.ar

Abstract. Deployment is the process by which a software system is transferred to a business client. A risk is defined as the likelihood for a loss to occur. In a software project, a risk might imply decreased quality of the software product, increased costs, a delay in project completion or a flaw, among others. A case study is developed with the aim to refine the set of risks. Furthermore, procedures are proposed for their prevention, mitigation and/or transfer for the software system deployment process. This article presents the results of a case study which analyzed the documentation related to deployment of functionalities in a bank's Human Resources Portal conducted by an Argentina-based software Small and Medium Enterprise (SME[1]).

Key words: software system deployment process, risk management, case study.

1 Introduction

There are various factors that can affect software projects, such as modifications in priorities and inadequate planning [1]. One of the most important factors might be unmanaged risks. A risk is the probability for a loss to occur. In a software project, such loss might take the form of decreased quality of the software product, increased development costs, a delay in project completion or a flaw [2].

A large number of projects lack formal approaches for risk management. The identification thereof usually depends, at an informal level, on the abilities and level of experience of software managers [3]. Although software risk management plays a key role in successful project management, it is usually not properly implemented in real world software projects, particularly in SMEs in Argentina [4].

[1] Presidencia de la Nación. (2020). https://www.argentina.gob.ar/noticias/nuevas-categorias-para-ser-pyme. Last updated on 07/06/2020.

Software system deployment is the phase of the development life cycle in which the software product is transferred to the client. The deployment process entails practices which tend to pose problems, such as the lack of components (generally external), incomplete downloads and faulty installations [5].

The problems that might arise in the deployment phase are transferred and they are eventually resolved during the maintenance phase. For this reason, an efficient software deployment process should save resources in terms of costs and effort [6].

Software deployment is usually conducted in distributed and heterogeneous environments, which add complexity, thus causing time consumption and additional costs [7]. Deployment entails a series of changes at several levels: processes, working methods, technology and organizational structure [8].

According to Reascos Paredes et al. [9], the main causes of technological risks include heterogeneous and incompatible infrastructure, SMEs' poor technological capabilities and competences, the complexity of these systems, and bad data quality and safety.

Forbes et al. [10] argue that the results of non-standardized and inadequate deployment practices are reflected in the information systems, which are difficult to maintain and operate.

This work presents the results of a case study aimed at refining (if necessary) the set of risks, as well as the procedures for their prevention, mitigation and/or transfer defined for the deployment process of software systems.

This article is organized as follows: related works are described in section 2; section 3 presents the set of risks for the deployment process; section 4 addresses the case study; and finally, section 5 presents the conclusions and future works.

2 Related works

A Systematic Mapping Study (SMS) was performed to build the state of the art on risk management for the deployment process of software systems [11]. After analyzing 100 primary studies, it was found that the most commonly used methodologies, methods and standards addressing risk management are CMMI [12], PMBOK [13] and SOFTWARE RISK EVALUATION [14].

To complement the SMS, a comparative analysis of the previously mentioned methodologies, methods and standards was conducted based on the DESMET method characteristics [15]. MAGERIT [16] was added to the comparison since it is one of the pioneering risk management methodologies [17].

The comparative analysis for the deployment addressed three dimensions: "Process", "Person" and "Product" [18]. After this comparative analysis, it was concluded that in the "Process" dimension all the methodologies, methods and standards analyzed address the risks for the deployment process. In the "Product" dimension, SOFTWARE RISK EVALUATION as well as PMBOK and MAGERIT include the risks of the deployment process while CMMI does not. Finally, in the "Person" dimension, none of the methodologies, methods or standards evaluated address the risks of the deployment process.

3 Risks of the deployment process

The activities and tasks considered for the definition of the risks of the deployment process are those stated in the technical process called “Transition” of the ISO/IEC/IEEE 12207:2017 standard [19]. This standard was chosen because it is internationally recognized. The activities and tasks are detailed in [20].

The risk classification used is the one proposed in [3], with adjustments made considering the evolution of software engineering in the last few decades and the deployment process of software systems. For risk weighting, the proposal established in the ISO/IEC 31010:2009 standard [21] is adopted, since it is one of the main international references in terms of risk management for the software industry.

The definition of risks was established considering a three-dimensional approach, given by the “Process” dimension, the “Person” dimension and the “Product” dimension [18]. The risks proposed for these three dimensions are described in [20].

4 Description of the case study

This section presents a detailed account of the case study following the guidelines proposed in [22].

4.1 Case study design

The main objective is to examine the feasibility of the application of a set of risks, as well as the procedures for their prevention, mitigation and/or transfer in the deployment process of software systems in a real environment with the aim to refine them (if necessary). According to Robson's classification [23], case studies fall under the scope of exploratory studies. We worked with documentation related to the deployment of capability deliverables for a bank's Human Resources Portal performed by an Argentina-based software SME.

4.2 Research questions

In order to address the objective of this study, the following research questions (RQ) are posed:

RQ1: How were risks managed during the activities of the software system deployment process (identification, analysis and severity)?

This question is intended to provide information about the risks encountered during the execution of the deployment process and the treatment provided by the consulting company in order to compare them with the proposal made.

RQ2: How can the software system deployment process be strengthened in this company?

This question is intended to determine the way in which the consulting company can enhance its deployment process. For this purpose, the identification of a set of

risks is proposed, along with the procedures for their prevention, mitigation and/or transfer.

4.3 Case and unit of analysis

This section describes the context, the case and the unit of analysis of the case study. According to Yin's classification [24], it is a holistic single-case study.

Context: the case study was conducted in a software SME located in the Autonomous City of Buenos Aires, with a total of 430 employees. This company develops customized information systems for clients of different industry sectors, including finance, automotive, pharmaceutical and banking. Its software projects combine agile practices with iterative life cycle development methodologies. Access was granted to the documentation of the project subject to an agreement not to disclose the name of the company and a commitment to inform about any findings and recommendations to be considered for deployment process risk management.

Case: deployment of deliverables for a Human Resources Portal conducted at a bank based in Argentina. It consisted in adding new capabilities, using a modular strategy. These were: integration with a new data source, publication of Application Programming Interfaces (APIs), integration with a distance learning portal, modification of the final user interface, new employee management alerts and notifications, appearance modifications to the application organigram, and modification to approval flows.

Unit of analysis: documentation related to the deployment of deliverables for a Human Resources Portal.

4.4 Preparation for data collection

A third-degree technique was used combined with an independent method according to the classification proposed in [25]. A template with a coding scheme made up of 3 groups was used. Each group coincides with the 3 activities of the technical process called "Transition" of the ISO / IEC / IEEE 12207: 2017 Standard [19] (A1 Preparation for deployment, A2 Deployment Execution and A3 Deployment Results Management).

Table 1 shows the traceability of the documents analyzed and the risks associated with each of the dimensions. The calculated risk weight is found in [20].

Table 1: Traceability of the documents analyzed for the case study (the defined risk coding scheme is detailed in [20]).

Documents/ Activities	A1	A2	A3
Risk monitoring spreadsheet	RProc6, RPers3 and RProd1	RProc10	RProd15
Progress Report	RPers4	RProc7 and RProd9	RPers13
Deliverable 1 - Closing report	RProd4	RProc8 and RPers9	RProc14, RPers15 and RProd13
Deliverable 1 -		RProd8	RProc15 and

Deployment report			RPers12
Deliverable 1 - Deployment Summary		RPers8	RProc11 and RProd12
Deliverable 1 - Deployment Tests Guide	RProc4, RPers2 and RProd5	RProd10	
Deliverable 1 - Deployment Test cases	RProc4, RPers2 and RProd3		
Deliverable 1 – installation scripts	RPers1 and RProd2		RProc12
Deliverable 1 – Work Plan	RProc5 and RPers5	RProd7 and RPers10	
Deliverable 1 – Installation Requirements	RProc1 and RProd6	RProc9 and RPers7	
Deliverable 1 - Deployment Completion report	RProc2 and RPers6	RProd9	RProc13, RPers14 and RProd14
Deliverable 2 - Closing Report	RProd4	RProc8 and RPers9	RProc14, RPers15 and RProd13
Deliverable 2 - Deployment Report		RProd8	RPers12
Deliverable 2 – Deployment Summary		RPers8	RProc11, RProc15 and RProd12
Deliverable 2 - Deployment Tests Guide	RProc4, RPers2 and RProd5	RProd10	
Deliverable 2 - Deployment Test cases	RProc4, RPers2 and RProd3		
Deliverable 2 – installation scripts	RPers1 and RProd2		RProc12
Deliverable 2 – Work Plan	RProc5 and RPers5	RPers10 and RProd7	
Deliverable 2 – Installation Requirements	RProc1 and RProd6	RProc9 and RPers7	
Delivery 2 - Deployment Completion Report	RProc2 and RPers6	RProd9	RProc13, RPers14 and RProd14
General Documentation	RProc3		RPers11 and RProd11

4.3 Analysis and Interpretation of Results

The results of the research questions defined for the case study are presented below:
RQ1: How were risks managed during the activities of the software system deployment process (identification, analysis and severity)?

Based on the documentation analyzed, it was possible to find flaws in the risk management proposed for the activities of the deployment process:

- Activity 1 (A1) – Preparation for Deployment: The deployment progress reports showed that, due to the few investments in technology made in recent years, the resources (hardware and basic software) assigned to the production environment did not comply with the minimum requirements requested by the

consulting company to carry out the deployment in accordance with the established work plan.

According to the deployment reports analyzed, the technicians (bank employees) did not have the knowledge and skills necessary for the correct deployment of scripts and monitoring of the guides sent by the consulting company. This is because the technicians who participated in the original deployment left the organization and were replaced by personnel with little technical or functional experience.

The general documentation of the project shows that the bank does not have an adequate personnel retention policy, which generates frequent rotation.

- Activity 2 (A2) – Deployment Execution: according to the progress reports of the deployment project, the technical flaws mentioned in the previous stage (separation of technical personnel with experience in the technologies involved and greater complexity of the product) generated friction between the consulting company and the managers of the bank. This was due to non-compliance with the deadlines established in the work plan, which ended up activating a penalty clause against the consulting company.

  During the documentary analysis, incomplete test plans and inadequate deployment metrics were found. According to the deployment completion reports, the consulting company had to face cost overruns for not having the document management procedures required by the bank in the contract and in corporate policy. In addition, it was necessary to add technical resources from the consulting company to address the lack of technical expertise of the bank's employees, who had to be trained to carry out future deployments.

  These technical drawbacks, added to a very demanding work schedule for internal reasons and needs of the bank (shown in the closing reports), were some of the causes that produced very important delays and friction between different sectors of the organization that even considered the cancellation of the deployment project on several occasions.

- Activity 3 (A3) - Deployment Results Management: problems with the software repositories (lack of necessary permissions, previous versions, lack of components, etc.), in addition to the low commitment and inexperience of the bank's technicians, generated multiple drawbacks during the deployment. These technical drawbacks strongly impacted on the quality of the final product and the satisfaction of the users who saw their productivity affected due to failures in the application's capabilities once the deployment was complete.

  In the deployment completion reports, it was also evidenced that there was a wrong dimensioning of the deliverables and that the necessary security tests were not carried out. This gave end users access to sensitive human resource information.

RQ2: How can the software systems deployment process be strengthened in this company?

Proper risk management minimizes drawbacks in the deployment process. In [20], the recommended procedures are presented to the software consulting company in order to prevent, mitigate and / or transfer each of the risks associated with the "Process", "Person" and "Product" dimensions.

4.4 Threats to validity

To analyze the validity of the study, the factors proposed in [25] were considered:

Construct validity. The results were obtained based on the documentary analysis of a set of risks for the process of deployment of software systems in a real context. This allowed us to answer the defined research questions, determining their relevance and suitability for the case.

Internal validity. The documentation used refers to a real case, a deployment of new deliverables for a Human Resources Portal performed in a bank in Argentina. In order to achieve greater precision and validity of the studied process, the need to combine the data source (project documentation) with other types of sources, such as interviews and / or focus groups to guarantee "data triangulation (source)", is recognized. Furthermore, the qualitative data collected and analyzed could be combined with quantitative data resulting from the project, thus ensuring a "Methodological Triangulation".

External validity. Carrying out a single case study may limit the generalizability of the results. However, a preliminary case study was conducted in [18]. These two experiences allow us to present results, which can be used by other researchers to carry out more studies with the same principles.

Reliability. The study data was collected and analyzed by the research group.

4.5 Lessons learned

- Method selection: a validation of a set of risks, as well as the procedures for their prevention, mitigation and / or transfer, for the process of deployment of software systems, was needed in a real environment, in order to refine them (if required). The results obtained allowed us to analyze the application of the set of risks defined in a real environment. Therefore, the method used is considered to have yielded the expected results.
- Data collection: although the documentation of the software system deployment process has been reviewed in order to analyze how the risks were managed, it is considered that the case could be strengthened if the data collected were complemented by another source or by quantitative data.
- Selected coding. The coding scheme selected for the design of the data collection and analysis template was adequate and allowed the systematic recording of risk information.
- Results report: Although the case is made up of two research questions, it is considered that the work carried out took into account an adequate level of detail for understanding the phenomenon under study.

5 Conclusions and future work

The results of a case study were presented to determine the feasibility of applying a set of risks, as well as the procedures for their prevention, mitigation and / or transfer

for the process of deploying software systems in a real environment. It consisted of the risk analysis of the deployment of new deliverables for a Human Resources Portal carried out by a software SME in a bank in Argentina. After conducting the case study, it is concluded that:

• The first question allowed us to identify shortcomings in risk management through documentary analysis. These shortcomings include the lack of specialization of project personnel, mixed interests between the intervening areas and non-compliance with requirements of the installation environment.

• The second question allowed us to design a set of recommended procedures (presented in section 4.3) for the company to improve its deployment process and to introduce good risk management practices for future software system deployments.

The lessons learned from the case showed that the research method was adequate to validate the proposal.

The following are identified as future works: (a) to validate the risk proposal for the software deployment process in different case studies in order to refine it. (b) To propose the use of the risks defined for the deployment of software systems, as well as the procedures for the prevention, mitigation and / or transfer thereof, by other professionals in the industry.

References

1. Charette R. Why software fails [software failure]. IEEE spectrum, 42(9), 42-49. (2005).
2. Dhlamini, J. & Nhamu, I. y Kaihepa, A. Intelligent risk management tools for software development. 33-40 (2009)
3. Jones C., Assessment and control of software risk. Yourdon Press (1994).
4. Liu D., Wang Q., Xiao J. The role of software process simulation modeling in software risk management: A systematic review. In Proceedings of the 3rd International Symposium on Empirical Software Engineering and Measurement. Empirical Software Engineering and Measurement, pp. 302-311 (2009).
5. Jansen S., Brinkkemper S. Definition and validation of the key process of release, delivery and deployment for product software vendors: Turning the ugly duckling into a swan IEEE International Conference on Software Maintenance, ICSM, art. no. 4021334, pp. 166-175. (2006).
6. Subramanian, N. The software deployment process and automation. CrossTalk, 30 (2), pp. 28-34 (2017).
7. Tyndall J. Building an effective software deployment process. In Proceedings of the 40th annual ACM SIGUCCS conference on User services, pp. 109-114 (2012).
8. Reascos I., Carvalho J., Bossano S. Implanting IT Applications in Government Institutions: A Process Model Emerging from a Case Study in a Medium-Sized Municipality. In Proceedings of the 12th International Conference on Theory and Practice of Electronic Governance, pp. 80-85 (2019).
9. Paredes I., Carvalho J. Research in Progress: Understanding the process of implantation IT Enterprise Applications in Small and Medium Enterprises (SMEs). In Atas da Conferência da Associação Portuguesa de Sistemas de Informação, Vol. 17, No. 17, pp. 270-283. (2017).

10. Forbes J., Baker E. Improving Hardware, Software, and Training Deployment Processes. In: Proceedings of 19th International Conference on Software Maintenance, pp. 377-380. IEEE, The Netherlands. (2003).
11. Ortiz F, Davila M., Panizzi M. y Bertone R. State of the art determination of risk management in the implantation process of computing systems. En las Actas del I Congreso Internacional sobre Avances en Nuevas Tendencias y Tecnologías (ICAETT 2019). Ecuador, Guayaquil Ecuador, 29 al 31 de mayo, pp- 23-32 (2019). ISNB 978-3-030-32022-5.
12. CMMI Institute, «Capability Maturity Model Integration,». https://cmmiinstitute.com Página vigente al 24/06/2020.
13. Project Management Institute. https://www.pmi.org/pmbok-guide-standards. Página vigente al 24/06/2020.
14. Software Engineering Institute, «Software Risk Evaluation Method» (1999). https://resources.sei.cmu.edu/asset_files/TechnicalReport/1999_005_001_16799.pdf
15. Kitchenham B., Linkman S., Law D.T. DESMET: A method for evaluating software engineering methods and tools. Keele University (1996).
16. Portal de administración electrónica, «MAGERIT v.3: Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información» 2012. Página vigente al 24/06/2020.
17. Ortiz F., Panizzi M., y Bertone R., Risk determination for the implantation process of software systems. En las Actas del XXV Congreso Argentino de Ciencias de la Computación - CACIC 2019. Universidad Nacional de Río Cuarto, 14 al 18 de octubre, pp-817- 825 (2019). ISBN 978-987-688-377-1
18. Panizzi M., Davila M., Hodes A., Vázquez P., Ortiz F., Arana F., Bertone R. Desafíos para la implantación de sistemas de software. En las Actas del XXII Workshop de Investigadores en Ciencias de la Computación (WICC 2020), El Calafate, Argentina 7 y 8 de Mayo de 2020. ISBN en trámite.
19. ISO/IEC/IEEE 12207:2017. Systems and software engineering — Software life cycle processes (2017).
20. Felipe Ortiz, Marisa Panizzi, Rodolfo Bertone. Appendix – Risk refinement in the deployment process of software systems: a case study. https://doi.org/10.6084/m9.figshare.12670967.v1.
21. International Organization for Standardization, «ISO/IEC 31010:2009». https://www.iso.org/standard/51073.html. Página vigente al 24/06/2020.
22. Runeson P, Höst M, Rainer A, Regnell B. Case study research in software engineering: guidelines and examples. Wiley Publishing, Hoboken (2012).
23. C. Robson. Real world research 2nd edition. Blackwell (2002)
24. Yin, R., Case study research: design and methods. 5th Edition. Sage Publications. (2014).
25. Lethbridge T., Sim S., Singer J., Studying software engineers: data collection techniques for software field studies. Empir Softw Eng 10(3):311–341 (2005).