An Agent-Oriented Approach for Assisting Risk Management in Software Projects

Thayse Alencar[1], Mariela Cortés[1], Nécio Veras[2], Lui Magno[1]

[1]State University of Ceará
Silas Munguba Avenue – Fortaleza — CE — Brazil

[2]Federal Institute of Education Science and Technology of Ceará
CE -187 —Tianguá — Ceará — Brazil

{thayse.alencar, magno.araujo}@aluno.uece.br, mariela@larces.uece.br,

necio.veras@ifce.edu.br

Abstract. *The management of software projects is a critical activity and susceptible to unplanned situations, commonly known as risks. Risks stem from a variety of sources, both external and internal to the project or organization; moreover, they can occur at any stage of the project life cycle. In this paper, we will present an approach that provides support for identification, analysis, response planning and risk control in software projects. For this purpose, an agent was developed and its behavior is based on metrics, change requests in the project, as well as the use of contingency reserves. The risk agent, ARis, was inserted into an existing multi-agent system (MAS) and in conjunction with all the agents, assists the prediction and mitigation of risks in software projects.*

1. Introduction

According to the annual survey conducted by the Project Management Institute, PM-SURVEY.ORG Edition 2014 , the most frequent problems in projects are: poor communication, failure to meet deadlines, inadequate definition of scope, constant changes in scope, insufficient human resources and risks not properly evaluated. However, the same survey shows that even international companies aware of this last problem, only 32% of them have a formal methodology (structured by policies, procedures and forms) for risk management; in other words, over 60% of the international organizations do not adequately address the risks of their projects or do it informally. These data emphasize the importance of proper project management, since problems such as those mentioned above compromise the goals of time, cost, quality and customer satisfaction, leading to risks that hamper the success of the projects.

A *risk* is an event of an uncertain condition that, if it occurs, will either have a positive or negative effect on the project objectives by: (1) threatening the *project* itself through the schedule, cost, and resources; (2) impacting the quality of the *product* that is being developed; (3) or affecting the *organization* at the business-level (Sommerville 2011) (Pressman 2011) (PMI 2013). In the organization or project, risks can emerge from both internal and external sources. As a result, both of these sources and triggers, which contribute to the progress of these events, require continuous monitoring and management.

According to Sommerville (2011), risk management is recognized as one of the most important tasks of project management. Knowing how to handle risks is one of the decisive factors for project success as it can compromise goals of time, cost, quality and customer satisfaction. In the current state of development of the area were found solutions that support the Risk Management in specific contexts or applying restrictions. Among the solutions found in the academy, many give only partial support, not including all RM (Fontoura and Price 2008) processes; others are completely theoretical (Rafele et al. 2005) or semi-automated (Knob et al. 2006) requiring still great effort from the manager; in addition to those developed for very specific contexts (Rad 2013).

Risk Management is an intrinsically complex activity, surrounded by challenges and uncertainties. Therefore, organizations must have a proactive and consistent approach to support risk management through the entire project life cycle (PMI 2013). In this context, intelligent agents have been demonstrated as a promising solution for supporting project management activities, due to their abilities: to detect and monitor changes in complex and highly dynamic environments; to reason about these changes; and to act proactively (Veras et al. 2015). In this paper, we propose a proactive and automated approach based on agent technology to assist the software project manager in the execution of the Risk Management processes (SEI 2010) (PMI 2013), regardless of the application domain.

This paper is organized as follows: Section 2 presents related works. Section 3 describes the proposed approach for risk management. Section 4 details the design and implementation of the multi-agent platform. Section 5 shows some findings obtained from the implementation of two processes of our proposal. Finally, conclusions and future work are presented in Section 6.

2. Related Work

Rafael et al. (2005) present a method for risk management in projects utilizing a (*Risk Breakdown Matrix* - RBM), formed by the combination of two other structures, the *Work Breakdown Structure* - WBS and the *Risk Breakdown Structure* - RBS. This strategy is useful for associating risks with the activities of a project. While the WBS defines the activities and packages of work in the project, RBS identifies possible sources of risk, and so the approach is able to perform a more robust analysis than the simple *Probability and Impact Matrix*.

Rad 2013 describes the *GOES-R Series RM*, a decision-making tool used to ensure safety and functionality of the *Geostationary Operational Environmental Satellite* - GOES system. The *GOES-R Series RM* has the risks in a *Risk Distribution Matrix* — a more robust version of the *Probability and Impact Matrix* — and positions them accordingly with the value of their risk exposure (E). When some significant change occurs in the project, the affected risks are updated and re-positioned in the matrix.

The use of project metrics can also be observed as a technique for supporting risk management processes. Fontoura et al. (2004) proposed an approach to risk prevention based on the customization of the organization's software process. The approach is oriented to defined metrics from the Goal/Question/Metric paradigm, and supports the Identification, Qualitative Analysis, Response Planning, and Risk Control processes. Considering the previously cited works, the approach also uses the *Probability and Impact*

Matrix technique in its most simplistic version to calculate the effect of a risk.

The usage of multi-agent systems supporting project management can be observed in some works. Nienaber and Barnard (2007) propose a *framework* to support all areas of project management proposed by (PMI 2013), by using specialized software agent technology, where each agent is responsible for a general task. The architecture of the *framework* integrates various multi-agent systems, and each of these systems is responsible for the processes of an area in management, while seeking to achieve a specific objective.

Another multi-agent platform for project management was developed by Veras et al. (2015) , which provides the monitoring and control of the project work and the integrated changes management. The approach applies the Aggregate Asset Management (AAM) and the Critical Path Method (CPM) (PMI 2013) to provide a consistent view of project progress, and assist the project manager in decision making. The agents are able to detect deviations during the execution of tasks and suggest corrective actions to reduce the negative impact of deviations.

By evaluating the available works, we notice that agent-oriented approaches in the context of software management projects, in particular for risk management, is a relatively new research field. The majority of works under analysis presented very limited mathematical formulations. In other words, the calculation of the risk exposure (RE) in these works excludes project or organization factors (e.g.: schedule, budget, staff) that contribute to the criticality of the risks. Moreover, it is noticeable a lack of automated support tool provided by these works.

3. Proposed Approach

To perform a robust analysis of the project's risks, the mathematical formulation developed in our approach takes into account these parameters: (i) the impact of each risk for the various project aspects (cost, schedule, scope and others); (ii) requested changes in the project; and (iii) the amount of available contingency reserve. Figure 1 shows the execution flow diagram of the macro-processes implemented by the risk agent ARis in the approach.

At the outset of the project, the properly identified and documented risks are incorporated into the internal state of ARis. Once the existence of risk factors jeopardizing the project is detected, the agent executes the process *Risk Analysis*, which consists of calculating the priority of each risk factor and updating its internal state. The occurrence of changes must be predicted during a project, but only formally approved change requests can be incorporated into the project's baseline (PMI 2013). To be approved, a change request needs to be evaluated because it might result in one or more modifications in the project attributes. Whenever the agent ARis is notified of any change request, it performs the *Simulation Environment* process to simulate the new state of the environment from the application of the change. If the change is approved by the manager, then ARis executes the *Updating Environment* process to update the information about the project's risks in its internal state. Soon after the processes of simulation or updating the environment, the ARis agent executes the process *Monitoring Project Metrics* in order to identify new risks or sources of risks.

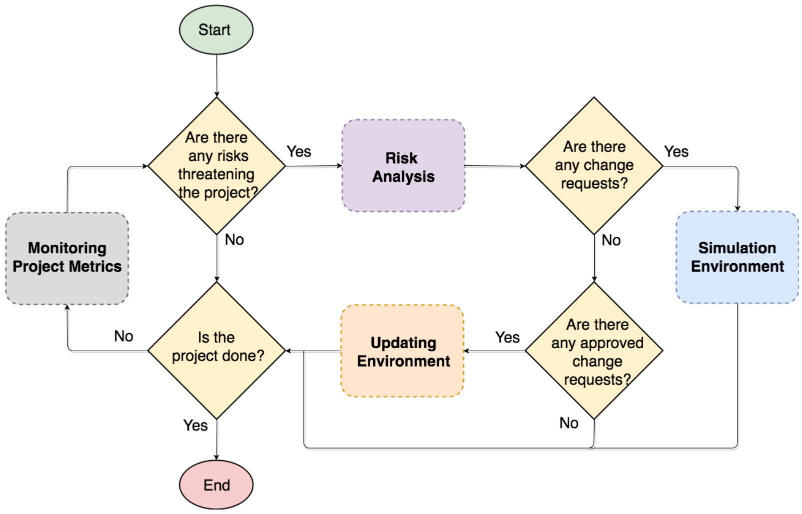

Figura 1. The flow diagram of the macro-processes performed by the ARis agent.

3.1. Approach Processes

3.1.1. Risk Analysis

The qualitative risk analysis process is executed in this approach through the *Risk Analysis* process illustrated in Figure 2. Let $LR = \{r_1, r_2, r_3, \ldots, r_N\}$ be the set of N risks that endanger a project P and let $CI = \{c_1, c_2, c_3, \ldots, c_M\}$ be the set of M attributes of the project affected by risks. Considering that each $r_i \in LR$ is able to simultaneously affect more than one attribute $c_j \in CI$, each risk r_i therefore has probability and impact values associated with each attribute c_j, respectively $P_{i,j}$ and $I_{i,j}$. Thus the estimated risk exposure for each risk r_i is given by Equation 1:

$$ER_{r_i} = \sum_{j=1}^{M} P_{i,j} * I_{i,j}, \tag{1}$$

where ER_{r_i} = total risk effect i; $P_{i,j}$ = risk probability i in an attribute $j \in CI$ and $I_{i,j}$ = risk impact i in an attribute $j \in CI$.

In this approach, the impact of each risk is measured from a scale of 1-5, where 1 is very low and 5 is very high. The probability is defined in percentage values from 0 to 100, representing the occurrence chances of the project events. An example of the application of Equation 1 for a set of risks LR is given by Table 1. In this example, the set CI is composed of the attributes Cost, Time, and Scope. The stored values in these columns of the table represent the risk impact in the attributes of the project. The columns labeled *PC*, *PT* and *PE* store the probability values of the risk in the same attributes of cost, time, and scope — in other words, the probability of a risk affects the attributes of the project. The row labeled *TOTAL ER BY AREA* store the total value of risk exposures by area of the project. The arrangement and analysis of the risks seen in Table 1 was adapted from (Rafele et al. 2005).

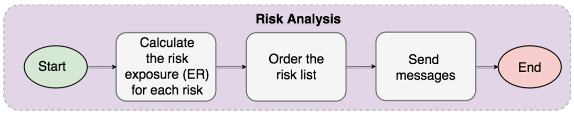

Figura 2. Process flow of Risk Analysis.

Tabela 1. Risk Exposure Matrix of Equation 1.

	Cost	Time	Scope	PC	PT	PS	ER	Classification
R1. Definition of Scope	-	-	IE = 3	-	-	30%	0.9	5
R2. Misunderstanding of the requisites	-	IT = 5	-	-	50%	-	2.5	3
R3. Incorporation of a new technology	IC = 5	-	-	70%	-	-	3.5	1
R4. Unrealistic schedule	-	IT = 3	IE = 1	-	90%	40%	3.1	2
R5. Unrealistic budget	IC = 4	-	IE = 1	30%	-	50%	1.7	4
R6. Development errors of the functions or interface	-	-	IE = 1	-	-	30%	0.3	6
TOTAL ER BY AREA	4.7	5.2	2.1					

3.1.2. Simulation and Updating the Environment

In the RM, project contingency reserves are used to respond to risks or to apply mitigation actions (PMI 2013), which may lead to changes in the project. The current approach uses the contingency reserve of time and cost both for the treatment of risks and possible changes in the project baseline requested by stakeholders, during the progress of the project. The amount of contingency reserve should be defined by the manager considering the size of the project, contract terms and the profile of the organization of the clients or investors. In this approach, the time and cost reserves of a project P are determined by:

$$RC_C = x\% * CP, \tag{2a}$$

$$RC_T = x\% * TP, \tag{2b}$$

where RC_C = Cost Contingency Reserve; CP = Total Cost of the Project; RC_T = Time Contingency Reserve; TP = Total Time of the Project; and x = Percentage established by the reserves.

The *Simulation Environment* process is based on the usage of the contingency reserve to measure the evolution of the risks. The Figure 3 illustrates this process. The use of contingency reserves for change requests has a direct impact on the progress of the project's risks, since the reduction of reserves implies the reduction of the resources to treat the risks. In other words, the smaller the amount of available reserves, the more critical the treatment of risks that could materialize. For this reason, the strategy of simulation of the change impact in the project environment before they are approved assists the manager in his/her decision making and provides clear and concise visualization risk progress in the project, hence simplifying the project execution of risk management processes in the organization.

The *Simulation Environment* process is executed every time that a change is solicited in an activity of the project. Let $A = \{a_1, a_2, a_3, \ldots, a_n\}$ be the set of N activities of the project P, an activity $a_i \in A$ is represented by a tuple (*id, title, estimatedTime, actualTime,*

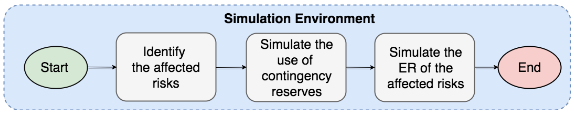

Figura 3. Process flow of Simulation Environment.

estimatedCost, actualCost, estimatedScope, actualScope). Thus a change in any activity $a_i \in A$ can affect its *estimatedTime*, *estimatedCost*, *estimatedScope*. In this approach, the variation in cost (MC_i) and time (MT_i) are defined by Equations 3a and 3b. Regarding variation in scope or any other attributes of the project, they are all mapped to cost and time ariations.

$$MC_i = CA_i - CE_i, \tag{3a}$$

$$MT_i = TA_i - TE_i, \tag{3b}$$

where MC_i = Change of cost in activity i; CA_i = Actual cost of activity i; CE_i = Estimated cost of activity i; MT_i = Change of time in activity i; TA_i = Actual time of activity i; and TE_i = Estimated time of activity i.

As stated earlier, the act of changing the project baseline may require the use of contingency reserves of time or cost. It is important to emphasize that the time reserves are only used for time changes in activities of the critical path of the project, hence these are able to impact the total duration of the project. The calculation of the amount of contingency reserves used for changes in the project P, at the same instant t, is given by:

$$URC_C = \frac{\sum_{i=1}^{k} MC_i}{RC_C}, \tag{4a}$$

$$URC_T = \frac{\sum_{i=1}^{k} MT_i}{RC_T}, \tag{4b}$$

where k = number of change requests at an instant t; URC_C = Use of cost contingency reserve; URC_T = Use of time contingency reserve; MC_i = Change requested i in cost at instant t; and MT_i = Change requested i in time at instant t.

Since the use of the contingency reserve has a direct impact on the progress of the project's risks, the amount of available contingency reserve increases or decreases the progress of the risks proportionally. Consequently, updating the progress of the affected risks requires updating their probability values. In this approach, the updating of the risk probability, i.e. the chance of affecting the attributes (time or cost) of the project is given by:

$$PC_i' = \begin{cases} URC_C * (1 - PC_i) + PC_i, & \text{if } URC_C > 0 \\ URC_C * PC_i + PC_i, & \text{otherwise.} \end{cases} \tag{5a}$$

$$PT_i' = \begin{cases} URC_T * (1 - PT_i) + PT_i, & \text{if } URC_C > 0 \\ URC_T * PT_i + PT_i, & \text{otherwise.} \end{cases} \tag{5b}$$

PC_i' and PT_i' are new probabilities of risk i affecting the cost and time of the projects, respectively. PC_i and PT_i are the initial probabilities of risk i affecting the the cost and time, respectively.

After updating the probabilities of the affected risks, during the *Simulation Environment* process, the agent recalculates the value of the same risks to close the simulation scenario and send a set of messages to the change requestor. The requestor, in turn, can analyze the scenario information and decide whether or not to approve the change in the project. After the *Simulation Environment* process, the ARis agent verifies for approved changes to initiate the *Updating Environment* process (Figure 4).

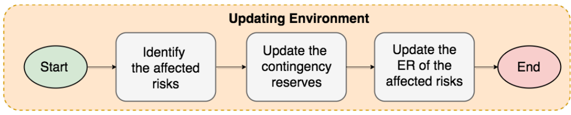

Figura 4. Process flow of Updating Environment.

3.1.3. Monitoring Project Metrics

In this approach, metrics are used for both triggering ARis' condition-action rules and assisting the manager to define the probability of risks in future projects. The flow of this process is shown in Figure 5. ARis' decision-making subsystem computes the set of metrics and executes actions based on the metrics' thresholds. Such actions might include: alert messages to the manager, prediction of new risks, suggestions of preventive/corrective actions that should be applied to the project, among others.

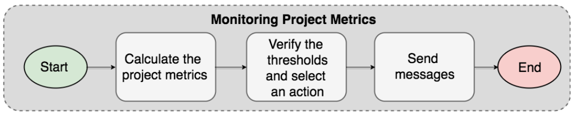

Figura 5. Process flow of Monitoring Project Metrics.

4. The ARis Agent and the Multi-agent platform

The implementation of the approach comes from the development and integration of the ARis agent into the multi-agent platform, which was proposed by (Veras et al. 2015). Originally, the platform includes three agents (AMon, ACon and AMud) that provides the monitoring and control of the project work and the integrated changes management through environment simulations of the project. Figure 6 illustrates the interactions between the agents and the components of the platform. In the referred illustration, the arrows indicate what actions the agents perform and what information the components send and receive.

AMon is responsible for monitoring the *Project Environment* with the goal of verifying deviations between the actual and planned project performance. **ACon** is in charge of the integrated control process proposed by (PMI 2013) and suggests preventive/corrective actions to the manager in order to reduce detected deviations. **AMud** is responsible for monitoring and control of change requests registered in the *Change Request Environment* through the SUT (Severity, Urgency and Trend) Matrix. **ARis**, the

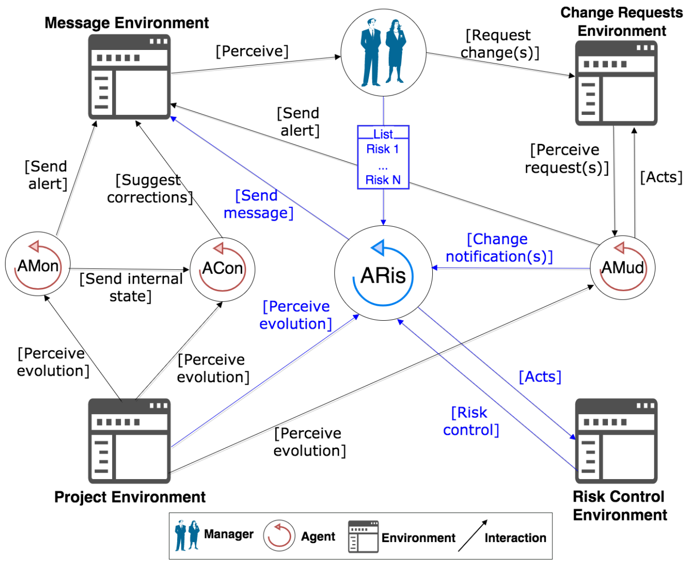

Figura 6. ARis and the multi-agent platform.

new agent, is responsible for the risk management of the project and acts based on the information obtained by its perceptions of the *Project Environment*. Moreover, ARis exchanges messages with the AMud agent, processes projects metrics, and performs the macro-processes of this approach, which were described in Section 3.1. This agent aims at contributing to the comprehensiveness of the multi-agent platform, by providing a robust analysis of the change requests managed by the AMud agent and assisting the project manager in decision-making.

5. Findings

Aiming at performing a feasibility study of the proposed approach, we developed the risk agent ARis and it has the macro-processes *Risk Analysis* and *Simulation Environment* fully implemented. Figure 7 illustrates the moment that ARis perceives the existence of three risks threatening the project, and it initiates the process *Risk Analysis* continuously until it receives a notification of a change request or observes a metric reaching a certain threshold.

At the instant 40, the ARis agent receives a notification of a change request made by the manager for activity I, requiring an increase of 11.9% in cost and 11.8% in time. Upon being notified, ARis starts executing the macro-process *Simulation Environment* and sending messages to the manager as seen in Figure 8. These messages consist of: variation in the cost or time of the activity (I in this case), the available amount of contingency reserve, and the list of affected risks including their new probability and values

As can be seen, the ARis agent automatically provides an anticipated view of the project's future state that might be considered by the project manager into his/her making decision process. Based on these results, we have confirmed our hypothesis that agent technology contributes to automated project management, mainly in proactive risk and change management. In conclusion, we claim that agent-oriented approaches are promising solutions that support the risk management processes regardless of the application context. To improve our findings, the macro-processes *Updating Environment* and *Mo-*

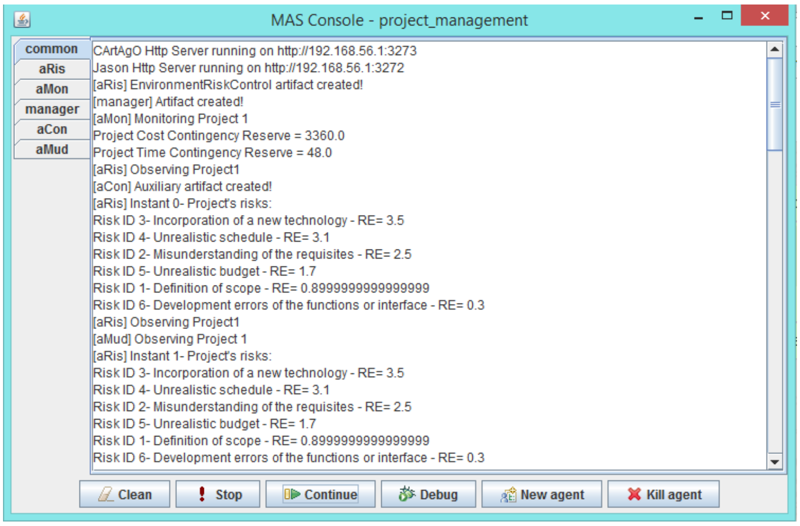

Figura 7. ARis executes the macro-process Risk Analysis.

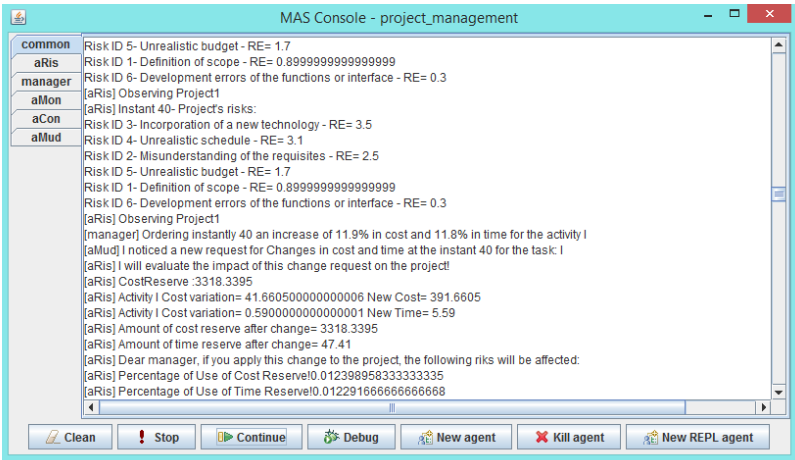

Figura 8. ARis executes the macro-process Simulation Environment.

nitoring the Project Metrics will be included in the release of the next version of this tool.

6. Conclusions and Future Work

In this paper, we present an approach that provides support in identification, analysis, planning of responses and controlling of risks in software project environments. Our approach incorporates a rich mathematical formulation that considers the usage of contingency reserve of the project to calculate probability and risk exposure, in addition, it uses metrics to reveal the progress of the project's risks throughout the project life cycle. As a result, the approach is expected to reduce the expenses of the organization via risk management, and to improve and control the project risk indices. The ARis agent is currently still being developed since the macro-processes *Updating Environment* and *Monitoring the Project Metrics* will only be included in the release of the next version. As soon we release the final version of this MAS, a real case study will be performed in the industry to refine our findings.

Referências

Fontoura, L. and Price, R. (2008). Systematic approach to risk management in software projects through process tailoring. In *SEKE*, pages 179–184.

Fontoura, L., Price, R., and Phil, D. (2004). Usando gqm para gerenciar riscos em projetos de software. In *Anais do XVIII Brazilian Symposium on Software Engineering*, pages 39–54. Brazilian Symposium on Software Engineering (SBES).

Knob, F., Silveira, F., Orth, A., and Prikladnicki, R. (2006). Riskfree: Uma ferramenta de gerenciamento de riscos baseada no pmbok e aderente ao cmmi. *V Brazilian Symposium on Software Quality (SBQS)*, pages 203–217.

Nienaber, R. and Barnard, A. (2007). A generic agent framework to support the various software project management processes. *Interdisciplinary Journal of Information, Knowledge & Management*, 2.

PMI (2013). *A Guide to the Project Management Body of Knowledge: PMBOK Guide*. Project Management Institute, Inc., 14 Campus Boulevard Newtown Square, Pennsylvania 19073-3299, Estados Unidos.

PMI (2014). Pmsurvey.org 2014 edition. Technical report, Project Management Institute.

Pressman, R. (2011). *Software Engineering: A Practitioner's Approach, 7th Edition*. The McGraw Hill.

Rad, A. (2013). Goes-r series risk management plan. Technical report, DOC, NOAA, NESDIS, NASA.

Rafele, C., Hillson, D., and Grimaldi, S. (2005). Understanding project risk exposure using the two-dimensional risk breakdown matrix. Technical report, Project Management Institute.

SEI (2010). *CMMI for Development, Version 1.3*. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA.

Sommerville, I. (2011). *Software Engineering, 9th Edition*. Pearson.

Veras, N., Cortés, M., Queiroz, A., and Souza, L. (2015). Abordagem proativa para a gestão integrada dos trabalhos de projeto. *Revista de Informática Teórica e Aplicada*, 22(2):166–180.