Definition and Assessment of On-Time Delivery in Rapid Software Development

Martí Manzano[1], Cristina Gómez[1], Claudia Ayala[1], Xavier Franch[1], Silverio Martínez-Fernández[2]

[1]Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
{mmanzano,cristina,cayala,franch}@essi.upc.edu
[2]Fraunhofer IESE, Kaiserslautern, Germany
Silverio.Martinez@iese.fraunhofer.de

Abstract. Rapid software development (RSD) is an approach for developing software in rapid iterations. One of the critical success factors of an RSD project is to deliver the product releases on time and with the planned features. In this vision paper, we elaborate an exploratory definition of the On-Time Delivery indicator in RSD and envisage a method for its estimation. This indicator will support decision-makers to detect development problems in order to avoid delays and to improve the planning and monitoring of the product releases.

Keywords: Rapid software development, On-Time Delivery indicator, Decision-making.

1 Introduction

Rapid Software Development (RSD) is the organizational capability to develop, release, and learn from software in rapid cycles [1]. One of the critical success dimensions of RSD projects is delivering their releases on time [2].

During RSD, a huge amount of project and development data (e.g., average speed to resolve issues) may be gathered from several repositories (e.g., JIRA). These data may be processed and analyzed with the purpose of turning into a meaningful and relevant indicator for giving decision-makers the view of how software development is going. One of the relevant information that may be assessed is the On-Time Delivery indicator. We define this indicator as the capability of fulfilling the issues (e.g. feature, improvement, project task, software bug or a custom issue) planned for a specific software iteration or release. The indicator provides useful information to decision-makers, as it can be used to estimate whether a team can complete a product iteration or release on time, or to discover untracked tasks, or development problems that may lead to delays on the software delivery. The main goals of this vision paper are:

- To present an exploratory definition of the On-Time Delivery indicator in RSD projects and to identify different factors affecting the indicator.

- To promote a preliminary version of a method to assess the value of the On-Time Delivery indicator based on Bayesian Networks. This assessment will allow decision-makers to detect specific planning, monitoring and development problems when the indicator has a lower value than expected.

Section 2 details the context in which the On-Time Delivery indicator is defined. Section 3 presents the definition and the factors influencing the indicator and Section 4 shows our vision to assess its value. Section 5 sketches the related work in the area. Finally, Section 6 summarizes the conclusions and identifies future work.

2 The Q-Rapids Approach

This work is carried out in the context of the Q-Rapids European project[1] that aims to improve the management of quality requirements in RSD processes. To achieve this goal, the project promotes a highly informative dashboard to support data-driven, requirements-related strategic decision making in rapid cycles. Such a dashboard is based on the extraction and analysis of data from project management tools, software repositories, quality of service and system usage. The analysis of these data permits to systematically and continuously assess software quality using a set of relevant quality-related indicators (e.g., customer satisfaction). Concretely, *Metrics* gather data from data sources using some software data collectors and are elaborated into *Product/Process Factors*, based on Quamoco [3] quality model, and ultimately aggregated into *Strategic Indicators*.

The research approach followed in the Q-Rapids project is based on action-research cycles [11] including the activities of identifying relevant problems in the context of the four industrial partners of the project and planning and performing actions to solve these problems. With this aim, we held several workshops and conducted semi-structured interviews to identify strategic and software product goals of the Q-Rapids industrial partners. The On-time Delivery strategic indicator emerged as a joint need of the partners.

3 On-Time Delivery Definition

In this section we elaborate the definition of the On-Time Delivery indicator and identify the factors influencing the indicator in the context of the Q-Rapids approach. We provide the definition of the indicator for the case of software development iterations but this definition may be easily adapted and applied to the case of software releases. For the definition of the indicator, we carried out the following activities:

- **Literature review:** A literature review was performed to get a consolidated background on the indicator and on its breakdown. Some of the topics reviewed were: time-to-market in software development, software development time and effort (and estimation of), effort estimation in agile software development and delivery capability in agile software development.

[1] www.q-rapids.eu

- **Tentative proposal:** From the literature review, a tentative On-Time Delivery indicator definition and its breakdown was proposed to the industrial partners.
- **Workshops:** Two series of workshops with the industrial partners were conducted to elicit particular understandings of their use cases and to get feedback on our tentative proposal, according to their subjective relevance.
- **Refining the proposal:** The feedback gathered from the workshops helped us to refine and build-up the On-Time Delivery indicator definition. We plan to refine the definition of the indicator during its deployment in the four use cases.

We define the On-Time Delivery indicator as the capability of fulfilling the issues planned for a specific iteration. The factors we have identified influencing the On-Time Delivery indicator are: Issues' Estimation Accuracy, Issues' Development Status, Issues' Due Date Compliance and Blocking. Table 1 shows the metrics for these factors, the data to gather for computing them, and their corresponding data sources. By measuring these factors, we can have a vision of the On-Time Delivery indicator through the Q-Rapids dashboard, and therefore decision-makers may perform specific actions to improve the delivery of the software. Next, we respectively explain the rationale of these factors.

Table 1. Factors affecting On-Time Delivery with their corresponding metrics

Factor affecting On-time Delivery	***Metrics and information to be gathered***
Issues' Estimation Accuracy	- *Accuracy of planning effort of issues*: Computed as the percentage of the difference between the planned effort of past closed issues and the actual tracked effort. - *Percentage of issues larger than the specified threshold*: From a threshold specified by the company in story points, hours or use case points, this metric gives the percentage of issues exceeding it.
Issues' Development Status	- *Ratio of the average past velocity and the theoretical velocity of the available units*: ratio between the averaged velocity of solving past issues and the theoretical velocity of the developers, taking into account their dedication - *Ability to resolve the remaining allocated effort* and - *Ability to resolve the remaining unallocated effort*: These metrics are computed based on the iteration being monitored, as aggregated information related to the development status, from the due date of the iteration, the estimated resolving velocity and the total allocated and unallocated remaining effort.
Issues' Due Date Compliance	- *Accuracy of planning Due Date's issues*: Computed as the percentage of the difference between the planned due date of past closed issues and the actual tracked issue closing date.
Blocking	Blocking's factors and metrics [4]

First, *Issues' Estimation Accuracy* indicates how reliable the current tasks' effort estimation is in terms of the past differences between planned efforts and the actual tracked ones. This factor can have an adverse impact on the On-Time Delivery if the past issues' effort has been under/overestimated, and influence the project manager or

the responsible to revise the effort estimation methodology, or to split-up issues in smaller sizes to facilitate more accurate estimations.

Second, *Issues' Development Status* refers to the development status in terms of the planned tasks, the assigned and unassigned effort, the average velocity of the development team and the remaining time until the iteration ending date. This factor enables the detection of low productivity, insufficient development resources allocation or high percentage of unassigned effort. For instance, if the factor has a low value, it could be fixed with actions to assign more resources to an issue.

Third, *Issues' Due Date Compliance* refers to the percentage of past due dates' compliance. If the percentage is low, it indicates that for some reason there have been problems to meet delivery dates in past issues, and the project manager or the responsible person should take measures to prevent it from happening again.

Finally, *Blocking* [4] refers to the blocking situations that arise when developing. Blocking situations increase waiting time, and therefore they are against the flow of constant delivery and can affect the on-time delivery of the planned content.

The presented factors are the ones we identified in at least one industrial partner. In general, when a company wants to measure the On-Time Delivery indicator for a specific software product, they can customize the definition of the indicator selecting the factors and metrics that can be computed from the company's available data.

4 On-Time Delivery Assessment

In this section, we provide a preliminary description of the method we envisage to assess the value of the On-Time Delivery indicator. In this method, we propose to use a mixed approach (expert-driven and data-driven) and to use Bayesian Networks (BNs) for the assessment because, as suggested in [5], it is a technique successfully used in software engineering that deals with uncertainty, it allows to create domain models being modelled using expert knowledge and/or data, and it enables what-if analysis. We propose the assessment method as follows:

Step 1. Building the BN's structure: To build the graph of the On-Time Delivery BN, some information will have to be elicited from the company for which the model is going to be build: the factors that will be parent nodes of the indicator, that could be the ones presented in Table 1 or a variation of these, and the categories for the parent nodes (e.g., each factor may be *Low*, *Medium* or *High*), and for the indicator node (e.g., On-Time Delivery indicator may be *Good*, *Neutral* or *Bad*).

Step 2. Gathering information from the experts: Once the structure of the BN has been built, the information of how the factors affect the indicator will be needed, in order to fulfill the Conditional Probability Table (CPT) of the On-Time Delivery node. This information shall be elicited from the experts of the company in specific workshops for this purpose. For instance, if all factors are *High*, without blocking, the expert probably will consider the On-Time Delivery indicator as *Good*.

Step 3. Validating and deploying BN: When the CPT of the BN is fulfilled, the BN will be ready to output assessments given a combination of factors' categories. At this point, we expect that some validation will be needed to assess the BN accuracy.

We contemplate using past data from the On-Time Delivery factors as inputs for the BN, to compare its output with the perception of the experts regarding the past status of the indicator and correct the possible deviations. After the validation, the BN will be ready for its deployment.

Step 4. Feedback gathering and refinement of the BN: Once the BN has been deployed in the Q-Rapids dashboard, it will give assessments regularly based on the status of its depending factors. In cases that the users (i.e. decision-makers) disagree with the estimated value, users will have the chance to provide feedback for the estimations, and that feedback will be used to revise and refine the BN periodically. Therefore, the model will always provide accurate estimations according to the perception of the experts of the company.

This assessment technique will not only enable the On-Time Delivery indicator estimation, but also allow what-if analysis of hypothetical scenarios.

5 Related Work

As far as we know, there is not a definition of On-Time Delivery indicator in the literature, even if it is a crucial context in RSD. However, a significant amount of studies has been dedicated to the definition of indicators or variables related to On-Time Delivery, as time-to-market in software development, effort estimation and delivery capability in agile software development (ASD). For instance, [6] defines On-Time completion as a dimension of software development performance and identifies related factors. An agile software estimation algorithm is proposed in [7] to make an estimation of the cost, size and duration of an agile project. Unlike the indicator we propose, On-Time completion and duration of a project are not defined in the context of iterations and factors related to the development status itself are not considered. In [8] the authors apply data mining techniques to build a delivery capability prediction model on a particular dataset, so their results and extracted variables are linked to the individual datasets used. They do not consider delivery capability as an indicator but as a response variable. In [9] and [10] a systematic literature review and a survey of the state of the practice in effort estimation in ASD are presented respectively, finding that subjective analysis is the most widely used estimation method. Common effort predictors are compiled, being the ones related to the size and complexity of the tasks and the skills and experience of the team as the ones that were observed in more studies. We do not explicitly include these variables related to the abilities of the team in the On-Time Delivery factors, because they cannot be easily estimated using the current project management tools. Furthermore, these team related metrics and more complex ones like productivity are implicitly taken into account with the *Issues' Development Status* factor.

The main difference between our work and those above mentioned falls on the fact that we are introducing a generic but customizable way to monitor the on-time delivery capability in RSD environments, so any company could adopt it with little effort. The factors and metrics used can be computed from the main project management current

tools like JIRA, Redmine or Gitlab. Furthermore, we envisage providing a highly customizable assessment method based on Bayesian networks.

6 Conclusions and Future Work

In this paper, we have defined the On-Time Delivery indicator as the capability of fulfilling the issues planned for a specific iteration, and identified the Issues' Estimation Accuracy, the Issues' Development Status, the Issues' Due Date Compliance and the blocking situations as factors that influence the iteration or release delivery of the software developed on time. Moreover, we provided our vision to assess the value of the indicator using BN techniques. As a future work, we are going to develop the complete method to assess the On-Time Delivery indicator. Moreover, we are planning to evaluate the effectiveness of our approach in the context of the Q-Rapids industrial partners and to iteratively adapt the indicator according to the beliefs of the industrial partners in order keep providing an appropriate and meaningful indicator.

References

1. L. Guzmán, M. Oriol, P. Rodríguez, X. Franch, A. Jedlitschka, & M Oivo. "How can quality awareness support rapid software development? - A research preview". REFSQ 2017, 2017.
2. T. Chow, D.-B. Cao. "A survey of critical success factors in agile software projects". *The Journal of Systems and Software*, 81, pp 961-971. 2008.
3. S. Wagner, A. Goeb, L. Heinemann, M. Kläs, C. Lampasona, K. Lochmann, A. Mayr, R. Plösch, A. Seidl, J. Streit, A. Trendowicz. "Operationalised Product Quality Models and Assessment: The Quamoco Approach". *Information and Software Technology*, 62, pp 101-123. 2015.
4. X. Franch *et al.*, "Data-driven requirements engineering in agile projects: the Q-rapids approach", *REW 2017*, 2017
5. E. Mendes, P. Rodriguez, V. Freitas, S. Baker, & M. A. Atoui. "Towards improving decision making and estimating the value of decisions in value-based software engineering: the VALUE framework". *Software Quality Journal*, pp 1-50. 2017
6. G. Lee, W. Xia. "Toward agile: an integrated analysis of quantitative and qualitative field data on software development agility". *MIS Quarterly*, 34, pp 87-114. 2010.
7. S. Bhalerao, M. Ingle. "Incorporating vital factors in agile estimation through algorithmic method". *International Journal of Computer Science and Applications*, 6, 2009.
8. M. Choetkiertikul, H. Khanh Dam, T. Tran, A. Ghose, J. Grundy. "Predicting Delivery Capability in Iterative Software Development". *Transactions on Software Engineering*, 14(8), 2017
9. M. Usman, E. Mendes, and J. Borstler. "Effort estimation in agile software development: a survey on the state of the practice". EASE'15, 2015.
10. M. Usman, E. Mendes, F. Weidt, and R. Britto. "Effort estimation in agile software development: a systematic literature review". PROMISE'14, 2014.
11. D. Avison, F. Lau, M. Myers, & P. A. Nielsen. Action research. *Communications of the ACM, 42*(1), 1999