Risk Analysis Tools for Managing Software Projects

NEVENA RANKOVIĆ, University of Novi Sad
MIRJANA IVANOVIĆ, University of Novi Sad

Risk analysis is a stage in the process of managing software projects, which largely affects the success of the project. It represents a complex and demanding process of planning, identification, analysis, response planning risks, monitoring, control, and reporting. Every software project, regardless of size, structure and project team is exposed to various types of risks. Risks can occur in the environment, but we have to accept the fact that they can not be completely eliminated. It is necessary to focus a lot of activities on risk reduction. The two most important features for improving the risk analysis process are: engaging professional resources, procurement and use of adequate software tools for risk analysis and management. This paper presents the results of comparative analysis of the 4 types of different risk management tools. We also proposed some specific measures for the selection and choice of the most appropriate of them in accordance with the requirements and needs of the user. Obtained results can help, not highly experienced software project managers, to select good risk management tool for leading their small and medium size software projects.

1. INTRODUCTION

The beginnings of risk analysis are related to the manual calculation of the necessary statistical data, forecasting, assessing and making numerous statements that required a lot of time and resources [DeMarco and Lister 2003]. Earlier, when appropriate tools are not used, a large proportion of the projects had been unsuccessful. The significance of risk management is gaining in importance rapid development of information technology, identification of project managers and software companies [Cooper et al. 2005]. The process of risk management is not sufficient and is not the only indicator of success, it is necessary to improve other elements and factors involved in software development. Introduction and use of software tools [Bajwa et al. 2008] facilitates the entire process and provide growing opportunities for analysis and risk management, and therefore the success rate of software projects is gradually increasing.

The aim of this work is to select some of currently available tools for risk analysis and management and compare their key functionalities. Further the idea is to identify key measures needed for the practical use of such tools, apply them on the selected tools and bring out conclusions and lessons learned that can help inexperienced, young project managers to select appropriate tools for their small and medium software projects [Demek et al. 2018].

In fact in the next section we review some of currently available tools, their basic characteristics, advantages and disadvantages based on which we make a selection of 4 distinctive tools [Bjelica 2017]. Additionally we present some key aspects of related work. In the third section of the paper we show details of analyzed important functionality of selected tools in two particular cases: two selected examples of software projects [Spenser 2008]. Fourth section presents performed analysis of tools in case of the selected examples and offers systematic discussion [Jones and Gallo 2007]. The fifth section brings concluding observations of performed analysis and the proposed measures

This paper is an excerpt from the master's thesis "Risk analysis in software project management" of Nevena Rankovic.
Author's address: Nevena Rankovic, Faculty of Sciences, Department of Mathematics and Informatics, Trg Dositeja Obradovica 4, 21000 Novi Sad, email: nevenar@dmi.uns.ac.rs; M. Ivanovic, Faculty of Sciences, Department of Mathematics and Informatics, Trg Dositeja Obradovica 4, 21000 Novi Sad, email: mira@dmi.uns.ac.rs.

In: Z. Budimac (ed.): Proceedings of the SQAMIA 2018: 7th Workshop of Software Quality, Analysis, Monitoring, Improvement, and Applications, Novi Sad, Serbia, 27–30.8.2018. Also published online by CEUR Workshop Proceedings (http://ceur-ws.org, ISSN 1613-0073)

[Carter et al. 2017] so that users have reliable cues when choosing and purchasing the right equipment.

2. RISK MANAGEMENT TOOLS ANALYSIS AND RELATED WORK

Our main motivation within our research activities was to select the tools for different purposes from the currently available and accessible tools for risk analysis in managing software projects. Through comparing their main characteristics and functionality (using two selected medium size projects as case studies) we would like to discover some of their key advantages and disadvantages. The need for such analysis is evident in order to select the appropriate tools to enable optimal performance for not highly experienced managers that lead some software projects. All components of risk that are characteristic of the management of software projects will be analyzed in the two concrete examples of projects. The results of comparative analysis will be presented later in the paper.

A software tool for project management is software that allows user to track the project from the beginning to the end [Peric 2012]. It provides continuously information on distribution of resources, assign tasks, flow of documentation, financial management, time and quality of the project [Udovicic and Kadlec 2013]. Tools for managing risks of software projects reduce the effects of surprises that make quick changes and emerging situations [Milentijevic and Zivkovic]. The main advantage of using such tools as accurate instruments is reducing the possibility of errors in an efficient and reliable way. Furthermore in order to identify possible risks it is necessary to be able to fast and adequately react when risks appeared. Software development process inevitably contains elements of uncertainty, defined as risks [Carter et al. 2017]. Successful completion of the project depends predominantly on the size of the risk corresponding to the project activity. In order to develop the activity in an appropriate manner, it is necessary to use a variety of technologically advanced tools, which require a certain level of knowledge for their use. Successful use of risk management tools is connected to defined phases such as: the identification, assessment, evaluation, control, reporting and management of all potentially major risks [Pijuk 2016]. Selected tool for risk management should enable the understanding, support and ways to mitigate risks that may jeopardize the success of the project. For software projects that have adequate time and cost constraints, it is necessary to invest adequate efforts in software development in which the process of risk mitigation is considered a central activity of management [Taylor 2004].

The goal of developers of the most risk management tools (to support software project management) is to implement them to be unique in terms of new features, greater efficiency and progress in the implementation of successful management and risk analysis [Pandian, 2007]. But some characteristics are considered to be included in every tool for risk management: continuous assessment of each risk and its way of dealing with the project, risk prioritization and implementation strategies for successful resolution of all software project risk [Brooks 1995].

Examples of the most commonly used tools for managing and risk analysis within software development are listed below [Ogunsanmi et al. 2011]:

- Risk Management Software by Resolver (see link 1.)
- Optial SmartStart by OptialTM (see link 2.)
- ProcessGene GRC Software Suite by ProcessGene (see link 3.)
- MasterControl Risk Analysis by MasterControl (see link 4.)
- A1 Tracker by A1 Enterprise (see link 5.)
- Checkit by Certainty Software (see link 6.)
- IsoMetrix by Metrix Software Solutions (see link 7.)
- Isolocity by Isolocity (see link 8.)
- Active Risk Manager by Sword Active Risk (see link 9.)
- Primavera Risk Analysis by ORACLE (see link 10.)

- RiskyProject Professional by Intaver (see link 11.)
- Full Monte 2016 by Barbecana (see link 12.)
- @RISK by Palisade (see link 13.)
- Project Risk Analysis by Katmar Software (see link 14.)
- Practical Threat Analysis by PTA Technologies (see link 15.)
- IT Risk Management Software by RISK VISION (see link 16.)
  and lot of others (see link 17.).

However each tool has its own advantages and disadvantages, and a special mode, which in the best and most appropriate approach shows the quality of managing risks within some particular software projects [Ogunsanmi et al. 2011]. Depending on the type of organization and the way it seeks to manage the risks different tools for risk management have to be assessed and finally the better one to be selected. For certain organizations particular operating mode of one tool may be useless, while on the other hand for another organization this simplicity of operation and display of basic statistical report are exactly what they need. Accordingly as a first step, it is necessary to determine the needs of the organization or team that will manage risks, then specify a group of tools that meet these requirements, and finally choose the appropriate one after performing adequate analysis of tools functionalities [Smith and Merrit 2002].

Generally speaking, still it is not easy to find a lot of researcher papers that explicitly compare essential functionalities of different risk management tools for software project management purposes. There are a lot of web sites (like: https://www.softwareadvice.com/risk-management/, https://www.getapp.com/finance-accounting-software/risk-management/, https://www.g2crowd.com/categories/governance-risk-compliance) which offer some kind of systematization of wide range of risk management tools but for wide range of different domains and purposes. They present usually long lists of various risk management tools and offer wide range of their general characteristics like, Rating of some stakeholders, Price, Platforms that support software, Deployment, appropriateness to different Business Size and so on. However they do not offer the clear comparison between them, between their functionalities and especially for specific purposes for software development. Indeed, much work has been done around the field of software development methodologies but the literature lacks such studies that conduct the comparative analysis in terms of risk management.

On the other hand there are some other studies like [Hijazi et al. 2012] that try to investigate the state of risk and risk management in the most popular software development process models (i.e. waterfall, v-model, incremental development, spiral, and agile development). They are also highly concentrated on risks but within specific software development process models and give no deeper comparison of different risk management tools.

The third group of research papers on risk management is oriented toward use of specific algorithms and techniques (recently more and more from Artificial Intelligence area) and also does not offer comparison of key functionalities of different risk management tools. Characteristic representative of such research direction is presented in the paper [Elzamly and Hussin 2014] where authors propose new mining techniques by which they can study the impact of different risk management techniques and different software risk factors on software analysis development projects. They used the fuzzy multiple regression analysis techniques with fuzzy concepts to manage the software risks in a software project and mitigating risk with software process improvement.

Having in mind abovementioned diversity of research and available literature on risk analysis and risk management tools our main intention was different. Our essential idea was to find easy way to help newcomers and not highly experienced managers in software project development and management to use some of “easy manageable” risk management tools. The intention was to help them to decide what tool to select for that purpose having in mind basic but essential functionalities and characteristics of such tools.

To achieve this goal we considered number of different risk management tools suggested by high number of experts. Based on their opinion we selected several appropriate tools which seems the most useful and with adequate functionalities for not highly experienced project managers.
Thus it is the main difference between our approach and approach of other researchers.

3. ANALYSIS OF SELECTED TOOLS

By examining the various functionalities of risk management tools (their complexity, applicability, reliability, price, implementation, compatibility with existing software projects, ease of use and similar) the organization and software project manager can, depending on the purpose and user requirements, to identify the most appropriate tool.

The findings presented in the paper and selected and analyzed tools are based on research activities conducted in the research center Petnica in Serbia during November and December 2017. The research included 56 participants/teachers of informatics in Petnica, who used the tools for risk analysis of small and medium scale software projects. The following results from the research activities were obtained and adequate and most appropriate tools are selected (Table I and Table II):

Table I. Percentage of commitment for a specific tool (Tools for medium scale software projects)

RiskyProject Professional 7	**Full Monte 2016**	**Other tools**
37.7%	**21.4%**	**42.9%**

Table II. Percentage of commitment for a specific tool (Tools for small scale software projects)

Practical Threat Analysis	**Project Risk Analysis**	**Other tools**
42.8%	**21.4%**	**35.8%**

Based on the available requirement specification, the user chooses a specific group of tools for analyzing and eliminating risks that can meet the demands of his/her company. The conducted abovementioned research and experiences of the previous users have motivated us for the selection of 4 tools and further analyzing their properties.

The first two selected tools, RiskyProject Professional 7 and Full Monte 2016, are more suitable for medium scale software projects. They will show the essential functionalities to which attention should be paid when choosing an appropriate tool.

Two other selected tools, Practical Threat Analysis and Risk Analysis Project, are tools that can be used independently as an auxiliary tool for efficient analysis of the most common parts of the project where risks may arise. To illustrate real-life process of assessment of several available/selected tools for risk management we performed following rather simple experiment: First step was to select four tools, Second step was oriented towards assessment of functionalities of selected tools. To perform the last task we also selected the two examples of software projects that can be seen as medium size projects. For our experiment following risk management tools have been selected: RiskyProject 7 is a complex and comprehensive software with greater opportunities for risk management than Full Monte, as it offers much more functionality such as risk identification, risk planning, risk assessment, risk response, monitoring and controlling risks and adequate reporting. Full Monte, on the other hand is more focused only on identifying and analyzing risk without planning and response to risks (to offer appropriate form of a report).

It may be noted that the significant lack of both tools is inability to export reports in a transparent Word or Excel documents, with which is easier and more manageable to inform organizations and end users. From the perspective of the user, RiskyProject gives the appropriate support and services, to the user and customer, and rather easy way to work with it. If we consider that the tool RiskyProject 7 is cheaper than Full Monte, we come to the conclusion that users may better take advantages of RiskyProject 7 and accordingly it is more recommendable.

Also from the users' point of view, RiskyProject gives the appropriate user manual and better customer service. If the organization is committed to risk analysis and obtaining reports in the allocation of resources and distribution of the same in the context of the development of a software project, it is far more efficient to use a specialized tools for this purpose, such as Practical Threat Analysis (PTA). In the case of PTA tool, it is possible to quickly and efficiently identify, mitigate or eliminate certain risks relating to resources within the project. If it is necessary in a short period of time to analyze one stage in the development of software product and thereby examine the possible risks of expected or unexpected costs, it is preferable to use tools specialized for this purpose, such as Project Risk Analysis.

Based on results of the analysis of all four selected tool, it can be conclude the following:
(1) RiskyProject Professional 7 is a tool intended for projects of medium and large scale. It contains all the features and necessary components for a complete and detailed analysis of the risks and all activities related to the risks of the project.
(2) Full Monte 2016 is a tool suitable for medium scale projects. The main its' lack is the impossibility of entering the response strategy of the risks in order to make adequate plans mitigate the potential risks of the project.
(3) Practical Threat Analysis is a tool that is suitable for risk analysis when allocating the resources needed to develop the project. It has the ability to define strategies of risk response and is suitable for rapid and brief analysis of parts of the project.
(4) Project Risk Analysis is a simple tool to analyze the cost of any phase of a software project.

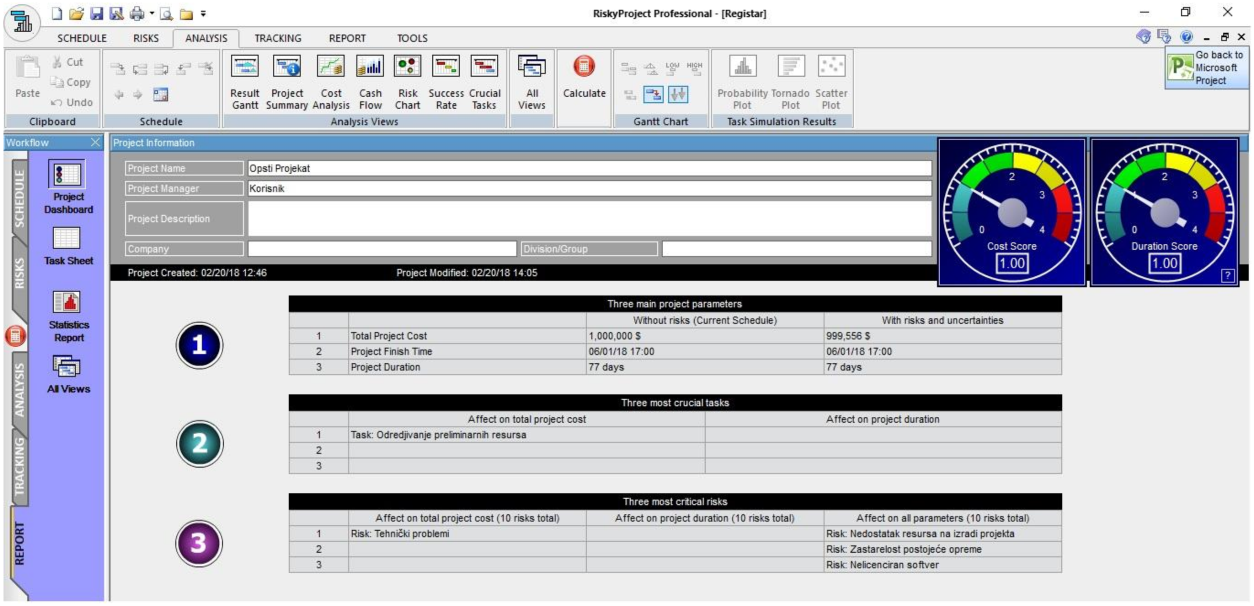
The general goal of these software tools is to increase efficiency to the development cycle of the project and to be available to all team members, so they can be more involved in the project.

4. DISCUSSION OF THE RESULTS

Two examples of selected medium scale software projects are completely prepared and realized using Microsoft Project tool. The first example is a general project called "OPRS" – organization of seminar. It was chosen as a good representative of general standard of most commonly used model of project development. It includes 10 elementary stages during the software life cycle with 75 activities, anticipated duration of 77 days and the indicative cost of one million RSD (Serbian currency). Eleven resources are assigned to the project, divided into two types (cost and work). For the purpose of comparative analysis was selected another similar case i.e. particular project example we will call it "Seminar organization for teachers - Petnica". It was also the project on seminar organization but for teachers in particular institution i.e. small research center in Serbian city Petnica. This project has 9 elementary phase with 55 activities, duration of 63 days and provided a budget of 685 000.00 RSD. In the total 33 resources have been allocated in this project, divided into two types (work and material).

After a detailed analysis of all the functionality of the first tool, RiskyProject Professional 7, in the first case of software project "OPRS", it has been obtained a final summary with all the uncertain parameters and the three most critical risks (Figure 1).

After a detailed analysis of all the functionality of the second selected tool Full Monte 2016, in the second example of the software project "Seminar organization for teachers - Petnica ", the "Tornado" diagram is obtained. This diagram generates the display of critical parts of the path and emphasizes the potential risks. Tabular view shows the following fields of interest: the remaining time duration of the task, the critical percentage of the task, the task-sensitive percent, and the sensitivity index with a small visual diagram, worst case and best case, end of the project with a small visual diagram (Figure 2).



Project Information

Project Name: Opsti Projekat

Project Manager: Korisnik

Project Description:

Company:

Division/Group:

Project Created: 02/20/18 12:46

Project Modified: 02/20/18 14:05

Three main project parameters

		Without risks (Current Schedule)	With risks and uncertainties
1	Total Project Cost	1,000,000 $	999,556 $
2	Project Finish Time	06/01/18 17:00	06/01/18 17:00
3	Project Duration	77 days	77 days

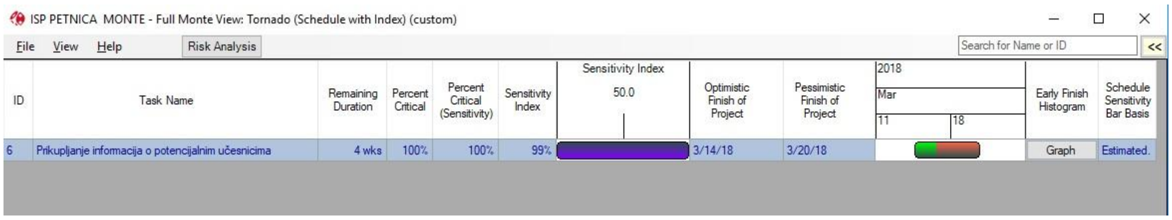
Three most crucial tasks

	Affect on total project cost	Affect on project duration
1	Task: Odredjivanje preliminarnih resursa	
2		
3		

Three most critical risks

	Affect on total project cost (10 risks total)	Affect on project duration (10 risks total)	Affect on all parameters (10 risks total)
1	Risk: Tehnički problemi		Risk: Nedostatak resursa na izradi projekta
2			Risk: Zastarelost postojeće opreme
3			Risk: Nelicenciran softver

Fig. 1. Screenshot of Final summary of the overall project "OPRS" with all the uncertain parameters and risks


ISP PETNICA MONTE - Full Monte View: Tornado (Schedule with Index) (custom)
File
View
Help
Risk Analysis
Search for Name or ID
<<

ID	Task Name	Remaining Duration	Percent Critical	Percent Critical (Sensitivity)	Sensitivity Index	Sensitivity Index 50.0	Optimistic Finish of Project	Pessimistic Finish of Project	2018 Mar 11 18	Early Finish Histogram	Schedule Sensitivity Bar Basis
6	Prikupljanje informacija o potencijalnim učesnicima	4 wks	100%	100%	99%		3/14/18	3/20/18		Graph	Estimated.

Fig. 2. Screenshot of customized "Tornado" diagram with the current schedule and the corresponding indexes of sensitivity

Third chosen tool Practical Threat Analysis is simple and has fewer features, designed for the analysis of the risk of a single phase of a software project, which is very important if it is necessary to quickly, efficiently and reliably analyze only the critical stages in the software project. It was chosen because of its simplicity of operations. Using tool PTA (Practical Threat Analysis) resources and risks that may occur during the development of software project seminar for teachers in Petnica are presented. This tool provides the ability to generate reports on self-assessment of performance completion phase of the project, after analyzing all the risks, resources and strategies for the project (Figure 3).

Fourth chosen tool Project Risk Analysis can be used exclusively for risk analysis of costs at any stage of a software project, which in turn allows the user to speed up and improve efficiency in their work. In this tool is presented one stage with six activities in the development of general software project "OPRS" phase "Design software project". The Project Risk Analysis tool can present the cumulative probability of the cost of the selected phase of the project presented at the selected interval of confidence (1% or 5%). The lowest costs at a confidence level of 1% in the aforementioned case, the same as the lowest costs at a confidence level of 5%. If we continue to look at the value, it can be concluded that increasing the confidence interval increases the costs as well (Figure 4).

Model Completeness

[illegible]

Model Entities

Threats	8
Assets	11
Vulnerabilities	8
Countermeasures	3

Threats

Assets

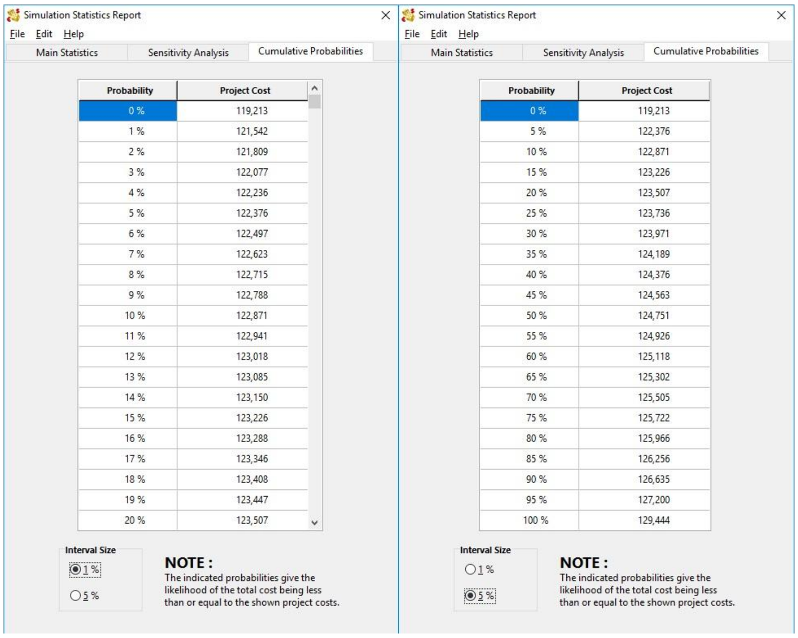
Fig. 3. Screenshot of self-assessment report of performance completion phase of the project seminars in Petnica

The most important advantage of these tools is that allows to an individual person or to whole project team to collect and process data, search, monitor and update them.

Finally after processing these two selected examples in four tools for risk management we are able to propose some significant key parameters i.e. measures to be considered when somebody would like to assess several risk management tools and select appropriate one. Proposed measures are illustrated in case of tools we selected and applied in our two chosen examples.

Advantages and disadvantages of assessed tools for medium scale software projects and for small scale software projects are presented in Table III and Table IV, respectively.

Tables show some of the tested functionality of represented tools. The complexity, applicability and reliability are high in case of RiskyProject Professional 7, while for the Full Monte 2016 they are medium. Practical Threat Analysis and Project Risk Analysis have low complexity, applicability very low, and reliability is high. Prices of procurement tools are high for the Full Monte in 2016, and in Risky Project Professional 7 they are medium, whereas in Practical Threat Analysis and Project Risk Analysis they are low. Implementation and price of above tools influence the selection and procurement of tools for risk analysis in developing countries such as Serbia. The table can be analyzed on the basis of elements that supports risk analysis, so Full Monte 2016, has no strategy response to the risks, while Practical Threat Analysis though is a tool designed for the analysis of one phase of the software project includes a plan and a strategy to risks.



Probability	Project Cost
0 %	119,213
1 %	121,542
2 %	121,809
3 %	122,077
4 %	122,236
5 %	122,376
6 %	122,497
7 %	122,623
8 %	122,715
9 %	122,788
10 %	122,871
11 %	122,941
12 %	123,018
13 %	123,085
14 %	123,150
15 %	123,226
16 %	123,288
17 %	123,346
18 %	123,408
19 %	123,447
20 %	123,507

Interval Size

● 1 %

○ 5 %

NOTE :
The indicated probabilities give the likelihood of the total cost being less than or equal to the shown project costs.

Probability	Project Cost
0 %	119,213
5 %	122,376
10 %	122,871
15 %	123,226
20 %	123,507
25 %	123,736
30 %	123,971
35 %	124,189
40 %	124,376
45 %	124,563
50 %	124,751
55 %	124,926
60 %	125,118
65 %	125,302
70 %	125,505
75 %	125,722
80 %	125,966
85 %	126,256
90 %	126,635
95 %	127,200
100 %	129,444

Interval Size

○ 1 %

● 5 %

NOTE :
The indicated probabilities give the likelihood of the total cost being less than or equal to the shown project costs.

Fig. 4. Screenshot of the cumulative probability of costs for the phase of the selected project presented at the selected interval of confidence (1% or 5%)

Table III. Tools for medium scale software projects

Characteristics	**RiskyProject Professional 7**	**Full Monte 2016**
Complexity	high	Medium
Applicability	high	Medium
Reliability	high	High
Price	medium	High
Implementation	application in medium and large scale software projects	application in small and medium scale software projects
Risk register	very good	Medium
Risk analysis plan	very good	Medium
Strategy for risk management	very good	None
Monitoring and control	very good	very good
Reporting	very good	very good

Table IV. Tools for small scale software projects

Characteristics	**Practical Threat Analysis**	**Project Risk Analysis**
Complexity	Low	Low
Applicability	very low	very low
Reliability	medium	Medium
Price	Low	Low
Implementation	good (but only for the resources in the project)	good (but only for the separate phases in project)
Risk register	good	None
Risk analysis plan	medium	None
Strategy for risk management	good	None
Monitoring and control	poor	Good
Reporting	good	Poor

5. CONCLUSION

The use of software tools facilitates the entire process and gives new opportunities for analysis and risk management, and therefore the success rate of software projects is steadily growing [Latkovic 2002]. Nowadays, use of tools for administration and the risk analysis are standards for all participants in the development of software products. Tools cost, but they ultimately yield results which are of greater importance than the actual costs of their purchases. The purpose of risk management is the introduction of effective control, prior to the limits established by the scope and responsibilities to achieve the appropriate level of security that will set goals to be accomplished. Successful risk management can provide [Carter et al. 2017]: better decision making, increase business efficiency, better forecasting and optimization of available resources, strengthening confidence in the organizational system, the development of good organizational culture.

The success of software companies today largely depends on the investment opportunities in quality management tools and risk analysis. Better and more efficient software are more capable of detecting, identifying, analyzing, reducing or eliminating the risks, and consequently the uncertainties the outcome of the entire project [Anshi et al. 2008]. Presented tools offer different opportunities to organizations and software project managers, in accordance to the size and costs of the project, and to the users to select similar tools according to their needs. By using appropriate risk analysis in the context of the appropriate tools, it is possible to direct the lower cost of the realization of a software project in the successful direction [Carter et al. 2017].

Analysis presented in this paper, can help smaller companies and not highly experienced software project managers, which have not yet begun to use these tools to analyze risks in their software projects to look at the basic and initial functionalities which are necessary to find and selected adequate tool for risk analysis [Milin et al. 2018].

Presented steps and lessons learned from assessment and evaluation of four risk management tools could be also used as a good and illustrative case study within a course on “Software project management” (such course is part of curriculum at our Faculty). It would illustrate to students the importance of risk management tools and their use during software project realization. Also they would learn what are necessary procedures and the selection of the essential characteristics that such tools should have.

REFERENCES

Anshi, X., Wei, S., Yan, D. 2008. Information risk management in financial service, 5th International Conference on Service Systems and Service Management. Exploring Service Dynamics with Science and Innovative Technology. 2.5, 082.

Bajwa, D.S., Lewis, L.F., Pervan, G., Lai, V.S., Munkvold, B.E., Schwabe, G. 2008. Factors in the Global Assimilation of Collaborative Information Technologies: An Exploratory Investigation in Five Regions. Journal of Management Information Systems. Volume 25, Issue 1, 131-165.

Bjelica, D. 2017. Integrated model for maturity assessment in IT project management, Ph.D. Dissertation, University of Belgrade, Faculty of Organizational Sciences, Belgrade. UC: 005.8. Avaible at: https://fedorabg.bg.ac.rs/fedora/get/o:16023/bdef:Content/get.

Brooks, F.P. JR. 2005. The mythical man-month. ISBN: 0201835959, USA.

Carter, D., Rogers, D., Simkins, B., Treanor, S. 2017. A review of the literature on commodity risk management, Journal of Commodity Markets, Volume 8, 1-17.

Cooper, D., Grey, S., Raymond, G., Walker, P. 2005. Project risk management guidelines, managing risk in large projects and complex procurements. Chichester: John Wiley & Sons.

DeMarco, T., Lister, T. 2003. Wlatying with Bears, Managing Risk on Sofware Projects, ISBN: 0-932633-60-9, New York.

Demek, K.C., Raschke, R.L., Janvrin, D.J., Dilla, W.N. 2018. International Journal of Accounting Information Systems, Volume 28, 31-44.

Elzamly, A., Hussin, B. 2014. Managing Software Project Risks (Analysis Phase) with Proposed Fuzzy Regression Analysis Modelling Techniques with Fuzzy Concepts. CIT, Volume 22, No. 2, 131-144.

Hijazi, H., Khdour, T., Alarabeyyat, A. 2012. A Review of Risk Management in Different Software Development Methodologies, International Journal of Computer Applications (0975–8887) Volume 45, No. 7, 8-12.

Jones, W., Gallo, A. 2007. A Process-Based Approach to Handling Risks, IEEE Journals&Magazines, Volume 9, Issue 2, 10 -

15.

Latkovic, M. 2002. Risk management: identification, measurement and control. Volume 26, Issue 2, 463-467.

Milin, D., Cosic, I., Moraca, S., Tesic, Z. 2018. Application of software tools for project management in Serbia, Department of industrial engineering and management, Faculty of Technical Sciences, Novi Sad.

Milentijevic, I., Zivkovic, B. Risk management in software development projects, Faculty of Electronic Engineering Nis, visited on 15.01.2018, https://www.scribd.com/document/52481649/Upravljanje-rizicima-u-projektima-razvoja-softvera.

Ogunsanmi, O.E., Salako, O.A., Ajayi, O.M. 2011. Risk Classification Model for Design and Build Projects, Journal of Engineering, Project, and Production Management, Volume 1, Issue 1, 46-60.

Peric, N. 2012. Methodology of IT project management and extreme programming, Master thesis, Faculty of Mathematics, University of Belgrade. dc.identifier.uri: http://hdl.handle.net/123456789/2427.

Pijuk, A. 2016. Project Risk Management and Comparison of the Two Risk Management Tools on the Project Development Example software, Faculty of Economics, University of Split. Avaible at: https://repozitorij.efst.unist.hr/islandora/object/efst%3A348/datastream/PDF/view.

Smith, P.G, Merrit, G.M. 2002. Proactive Risk Management, New York: Productivity Press. ISBN-10: 1563272652.

Spenser, M. 2010.Understanding the Software Development Process, available at projectsmart.co.uk.

Taylor, J. 2004. Managing Information Technology Projects: Applying Project Management Strategies to Software, Hardware, and Integration Initiatives, Amacom Books, ISBN-10: 0814408117

Udovicic, A., Kadlec, Z. 2013. Analysis of risk management, University Sibenik, UC: 658:330.3. Avaible at: https://hrcak.srce.hr/file/175103

1. https://www.resolver.com/apps/enterprise-riskmanagement-software, visited on 15.01.2018.
2. http://www.optial.com/solutions/ehs-smartstart, visited on 09.01.2018.
3. http://processgene.com/solutions/grcsoftware, visited on 12.01.2018.
4. https://www.mastercontrol.com/risksoftware/management_applications.html, visited on 5.01.2018.
5. http://www.a1enterprise.com, visited on 4.01.2018.
6. http://www.certaintysoftware.com, visited on 23.01.2018.
7. https://www.isometrix.com, visited on 18.01.2018.
8. https://www.isolocity.com, visited on 3.01.2018.
9. http://www.sword-activerisk.com, visited on 10.01.2018.
10. https://www.oracle.com/applications/primavera/products/risk-analysis.html, visited on 17.01.2018.
11. http://intaver.com/products/riskyproject-professional, visited on 22.01.2018.
12. https://www.barbecana.com/full-monte, visited on 12.01.2018.
13. https://www.palisade.com/risk, visited on 19.01.2018.
14. https://www.katmarsoftware.com/pra.htm, visited on 21.01.2018.
15. http://www.ptatechnologies.com, visited on 14.01.2018.
16. https://www.resolver.com/risk-vision-information-security-software/, visited on 10.01.2018
17. https://www.capterra.com/project-management-software/, visited on 10.01.2018.