An Extended Agile Software Development Project Budget Model

Osaki Thom-Manuel[1], Chidiebere Ugwu[2], Laeticia Onyejegbu[3]

[1, 2, 3] Department of Computer Science, University of Port Harcourt, Port Harcourt, 500102, Nigeria

[1]osakimil@yahoo.com, [2]chidiebere.ugwu@uniport.edu.ng, [3]nneka2k@yahoo.com

ABSTRACT

The aim of this paper is to develop an extended agile software development project budget model for the incorporation of an explicit risk management practice in the agile software methodology. This became necessary since the importance of managing risks explicitly in the agile software development methods is increasingly being stressed as seen in the literature reviewed. However, the knowledge of the constituents of explicit risk management is not streamlined yet and consequently its cost is not known. To achieve the aim of this paper, the constituents of an explicit risk management process were first derived. This was followed with development of Use Case model to enable the researcher identify the various requirements that would be involved in developing the model. The development of the extended model was then made with the aid of the use case model. The model, though theoretical, will give insight to stakeholders, developers and business analysts on the budget requirements of explicitly managing risk in the agile software development projects.

Keywords: *Initial Agile Software Development Project Budget, Explicit Risk Management Cost, Agile Risk Information Track Sheet, Risk Management- related Training Budget.*

1. INTRODUCTION

Software risk management is the systematic process of identifying potential risks, analysing and taking precautionary measures to reduce/eliminate its occurrence or impact. Risk management is continuous and a must-follow practice in every software development process and project intended succeed. As a result, it is considered a standard practice in the software industry (Islam, 2009; Kontio, 2001; Nyfjord & Kajko-Mattsson (2008); Wiegers, 2007). Agile software development refers to software development methodologies that adhere to the concepts of the agile manifesto under which requirements and solutions evolve through collaboration between self-organizing cross-functional teams in an iterative and incremental way. (Collier, 2011; Caseley, 2014; Deloitte Development LLC, 2017; Agile Alliance, 2015). Because of its iterative nature, agile development methodology is considered capable of meeting with the constantly changing requirements early delivery and high quality software need of customers. This, however, is achievable especially in small software projects as the specified agile principles is suitable for the development of only simple and small software projects having small teams (Highsmith & Cockburn, 2001 ; Alharbi, Rizwan & Qureshi, 2014). More so, even agile proponents agree that agile is suitable for small low-risk projects but formal risk management is necessary if it must be used for 'other' projects (Cohn, 2014). This clarifies the fact that some modifications have to be made to utilize agile methodologies for large-scale software projects development. Although, agile methodologies have gained much recognition by many in the industry, there exists little evidence that the practices they recommend reduce risks (Dhlamini, Nhannu and Kachepa, 2009). Yet, it is believed by the executors of agile methods that agile is risk-driven and do not need any form of explicit management of risks (Beck, 2004); Scrum, 2003). As a result there is no definite consensus on the need for explicit risk management within the Agile method (Veethil, 2013).

Also, risk management is a project on its own and as such, integrating it into the software development project is considered a project within another project. This, in turn, means that risk management has its own cost. Accordingly, an explicit risk management practice with risk management cost estimation suitable for agile development setting will be helpful in mitigating risks even in large software projects since such practice is proactive, continuous and systematic. However, despite the need for explicit risk management in the agile methods, stakeholders are doubtful of explicit risk management constituents in terms of cost when compared with managing risks implicitly. They fear that explicit risk management might involve more fund than implicit management of risks. This is clear from a statement by Cohn (2014), that explicitly managing risks is costlier than implicitly managing risk. This statement is without proof as there are currently no known ways to quantify risk management cost as a whole. This paper provides clarification on the software project budget

composition with explicit risk management incorporated into the agile settings. This will enable stakeholders and development teams to distinctly know what and what will be involved to implement risk management explicitly in the agile software projects. With this, knowledge of the financial involvement of implementing risk management in the agile methods explicitly can be easily estimated, thus relaxing the fears entertained by organizations about cost that will be accrued as a result of managing risk explicitly.

2. REVIEW OF RELATED WORK

Though few studies exist on risk management in the agile software development methodologies, they laid no emphasis on the budgetary involvement/composition in accomplishing explicit risk management; these rather introduced various tools such as risklist, risk burn charts, models and frameworks to follow in order to perform risk management. Meanwhile, first and foremost, stakeholders (including the agile team themselves) need to know the financial involvements in order to accept doing formal risk management bearing in mind that risk management has associated cost. Some of the existing studies include the study by Khatavakhotan, Hashemi and Ow (2011), which proposed a mathematical Risk Management model for iterative IT projects based on the smart database. The model was based on the Agile software development methodology. It consists of nine steps, namely Risk Identification, Analysis, Risk Cost Evaluation, Total Risk Amount Calculation, Total Risk Calibration, Requested Risk Budget, Real Risk Budget, Risk Removal Process and Iteration Results. Some case tools, namely Agile Risk Information Sheet (ARIS) and a relational DBMS, were utilized. Agile Risk Information Sheet is used for taking down and registering related information with different risks in an Agile iteration. This is an ideal model for use in an iterative model such as agile method. Apart from being detailed, it incorporates risk cost estimation tool. This is good in estimating what it takes to rework task(s) affected by a particular risk if the risk eventually occurs. Nonetheless, risk cost is but a sub-cost of an entire risk management cost and does not represent a total cost of explicit risk management cost.
Matkhour, Assassa and Baihan (2008) proposed a risk management tool specifically for extreme programming using the Microsoft Excel application tool which was arrived at after a pilot study. First of all, the pilot study to know the relevance of the twelve XP practices was conducted. This consisted of 25 questions that were directed at many channels including some software developers working with 27 Toronto groups, computer science students from the University of Manitoba and an open call on a number of websites. Inputs in the tool included project budget, risk management budget, cost of controls, Single Life Expectation (SLE). Annualized Loss Expectancy information entered is used to analyze the priority practices as to decide which of the 12 XP practices to deal with first. Thus, this tool helped to analyze risk associated with the twelve practices of extreme programming. Although, it formulated ways of calculating risk control cost and its benefits, the risk management budget as calculated here does not include all budgets involved in risk management. Scrumology (2014) suggested ways to estimate total cost of an agile software project. The cost referred to here does not include explicit management of risk. Likewise in the study by Madden (2014) steps to arrive at budgeting an agile software projects were made. However, explicit risk management budget were not considered. From literature reviewed on budgeting for agile software projects, none specified the inclusion of budget for explicit risk management. Meanwhile, some form of explicit risk management in medium to large-scale agile software projects are inevitable if such projects must meet the targeted requirements at the scheduled time within the right budget. Consequently this paper will contribute to knowledge in this regard by identifying the various expenses that will emerge as a result of explicit risk management and develop a budget model for agile projects with explicit management of risk incorporated.

3. THE EXTENDED NEW AGILE SOFTWARE DEVELOPMENT PROJECT BUDGET MODEL

An extended New Agile Software development project budget model is proposed. It consists of all budgetary steps and activities that are needed to be performed during an Agile software development project with explicit risk management incorporated. Ordinarily, an agile project budget will include the software development project cost (without explicit risk management), but with a formal method of managing risks, other activities that attract cost are performed. Khatavonkotan et. al in their study had identified one cost aspect- the risk cost. This will be part of the budget consideration in the proposed system, hence making this system an extended form of the existing composition.

3.1 The Extended Agile Software Development Project Budget Composition

By our own contribution, the extended agile software project should consist of two budgets, namely the initial software development project budget and the risk management budget.

i. Software Development Budget is the budget made without consideration for explicit risk management in an agile software development project. This consists of the usual Agile software development budget made at the beginning of software project after the need of the stakeholder had been related to the agile consulting group. Usually it is a high level budget obtained using a top- down approach to quickly get major features of the software needs.

ii. Risk Management Budget consists of all it will take to actualize or carry out formal risk management in a particular project. It includes RiskCost and OtherExpenses. RiskCost entails what it will take (manpower effort) to rework or correct a risk situation if the risk occurs, but risk management is more than that as it does not involve only the manpower effort in reworking risks in developing the software but also includes OtherExpenses, namely infrastructural expenses, training expenses and risk repository system expenses as well as risk manager salary.

- OtherExpenses of Risk Management

    Infrastructural Expenses: this refers to devices/accessories required to undertake risk management process. Because risk management is a process of its own and a formal practice need to be separately handled even if it is to be integrated into the agile development, risk monitoring and repository need to be implemented using a separate computer system, either a desktop or a laptop. For instance, from the beginning of the risk management process, identified risk and other related details are entered into the risk repository via the risk application which is in the system. This includes a separate computer system, backup device (secondary storage or cloud), printer, its maintenance and accessories. In this study, the average lifespan of computer devices is taken to be 3-5 years Koble (2015), and so infrastructural expenses are divided into three years and the result of a year is further shared amongst the major projects of the year using their project size to determine the shared value per project. Thus, the bigger the size of a project, the bigger the cost of risk management apportioned to it. Each software development project does not involve the purchase of new infrastructure. Such devices are purchased only when they are faulty, other times repairs are made and possibly including purchase of component parts.

- Risk Management Repository Budget is budget for developing and maintaining the risk repository system. The repository should be automated and should consist of the Agile Risk Management Application and the database. Identified risk details are entered into the repository system for record and computation of risk management cost purposes. Thus such a repository is very useful for quick computations and could serve as historical data for reference use.

- Risk Management- related Training Budget includes training cost for agile team on the use of the risk management system. This training cost may come up if, for instance, a new team member is engaged, especially when a team member leaves the firm. Generally, it includes all risk management-related training for members of the Agile Team and the Risk Manager where the need arises.

- Risk Manager Salary

    This is money expended for the payment of the services of a Risk Manager who dedicatedly monitors and carries out risk management activities. Formal implementation of risk management goes well with the engagement of a risk manager who does no other thing but to ensure that identified risk information, monitoring and repository tasks are efficiently and effectively carried out. It is very important to assign risk database management to a risk manager to avoid data duplication and to provide reliable historical data. A risk repository that is visible to all is usually more political than useful (Software Engineering Institute, 2008). The Risk Manager need full concentration on his task, especially when the software development is done on a large scale. Risk Manager remuneration can be fixed or derived by calculated hours of work and rank. Also, the risk manager can be a team member dedicated to managing risk management issues and given all necessary training. In that case he is part and parcel of the team and may not be given a separate remuneration but paid as a team member. However, he does any other job when no risk management work is available. Figure 1 depicts the extended agile software management budget composition.
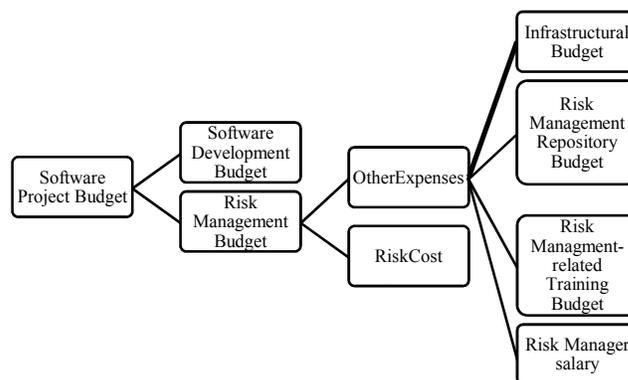

Fig. 1. The Extended Agile Software Project Budget

Risk Management Expenses sheet					
Infrastructure					
Infrastructure ID: Date:	Category	Description	Cost	Maintenance charge(*if any*)	Amount
	☐ urchases ☐ Repairs ☐ Services Service Provider:				
	Reason:				
Risk Repository					
Risk Maintenance ID. :	Category	Description	Maintenance charge(if any)		Amount
	☐ Initial Development ☐ Maintenance				
	Service Provider:	Reason:			
Training					
Training Ref. ID:	Category	No Of Staff	Training Description	Cost/ Staff	Amount
	☐ New Staff ☐ Old Staff ☐ General Duration:				
Risk Manager Details					
RiskManager ID:	**Category**	**Name**	**Staff Level**	**Man Effort**	**Remarks**
	☐ New Staff		☐High-Skilled	Fee/hr:	
	☐ Development Team Member		☐ Skilled ☐ Semi-Skilled	Fee/day:	
				TotalPay/Mth:	

Fig. 2. Risk Management Expenses   History Sheet (RIMEHS)

3.2 CASE Tools for Extraction of the Data

CASE tools, namely Agile Risk Information Track sheet (ARITS) – modified and adopted Risk sheet of Khatavokotan el al and Risk Management Expenses History
Sheet (RIMEHS) are two sheets developed for extraction of risk-related information from agile team.

3.2.1 Risk Management Expenses Sheet (RiMEHS)
is a special CASE tool used to record all infrastructural

expenses, risk repository development and maintenance expenses, training costs and risk manager remuneration. It is used at any time in the risk management stages when the need arises. The Risk Management Expenses sheet is shown in Figure 2.
Explanation of the Risk Management Expenses History Sheet
The Risk Management Expenses Sheet consists of Infrastructure, Training, Risk Manager's Remuneration and Risk Repository expenses sections.
Under Infrastructural expenses, three categories of expenses fields with check boxes, namely Purchases, Repairs and services are provided. At any particular point in time, the check box of a category concerned is checked. Infrastructural ID is entered in the space provided to uniquely identify the very infrastructure of concern. Infrastructural Purchase expenses include expenses relating to equipment like computers and their accessories as well as ink, toners, paper, etc .
Expenses relating to repairs include those of equipment used for risk management implementation. Description of such expenses is entered.
Cost field used to capture cost per equipment or accessories cost.
Amount field is used to enter total amount of purchased equipment, repairs or services made. Maintenance charges are only applicable to charges for services and repairs. Service provider includes the technical people engaged in the service being recorded.
In the Risk repository section, initial development check box is checked only when the expenses recorded are for the initial development of the repository. Maintenance check box is checked if only maintenance is performed on the repository. The service provider is the one who either develops the risk repository or does some maintenance work on it and the reason field is to state why that expenses are made.
The training section is used to record all training expenses. Fields in this section are self-explanatory.
Also, the Risk Manager's salary details are recorded in the last section. Risk Manager can be a new or old staff. The check box of the relevant field is checked. Depending how payment is made, relevant fields as in fee/hr (fee per hour), fee per day and/or per month, is filled.
The name field captures the Risk Manager's Name.
Staff level check boxes are also checked for the relevant field according to the experience of the risk manager.

3.2.2 Agile Risk Information Track Sheet (ARITS)
is the sheet used to record each identified risk details during in each iteration. ARITS is shown in figure 3. It is a modified form of ARIS of MRMM which is a Smart

database system as it can be used to capture all required risk information details.

Agile Risk Information Track Sheet (ARITS)				
Project Name:		Iteration Number:		
Risk ID:	Date: / / Time: min	Category	Activated date from / / To: / /	Record No:
Risk Description:				
No. of Tasks Affected: Risk Removal Necessity Level:				
Task No.: Task Description: Planned Hour: (mins) Overtime: (mins) Task Relevancy Level: Involved Person: Rank: High skilled person (hs)☐ Normalized Coefficient : Skilled person (s) ☐ Normalized Coefficient : Semi-skilled person(ss)☐ Normalized Coefficient : Total hours:				
Task No.: Task Description: Planned Hour: (mins) Overtime: (mins) Task Relevancy Level: Involved Person: Rank: High skilled person (hs)☐ Normalized Coefficient for: Skilled person (s) hours☐ Normalized Coefficient for: Semi skilled person (hs)☐ Normalized Coefficient for: Total Hours:				
Task No.: Task Description: Planned Hour: (mins) Overtime: (mins) Task Relevancy Level: Involved Person: Rank: High skilled person (hs)☐ Normalized Coefficient : Skilled person (s) hours☐ Normalized Coefficient : Semi-skilled person(ss)☐ Normalized Coefficient : Total hours:				
Mitigation Activities:				
Monitoring Activities:				
Contingency plan:				
Current Status:				
Involve People's ID:				
Comments:		Name ID AND Signature:		

Fig. 3. Agile Information Track Sheet (ARITS)

Explanation on the Agile Information Track Sheet Fields
The Project Name is the descriptive name of the software system being developed.
The Iteration Number is an identifier of each iteration in the software development project or system. For example, the first and second iteration is identified as iteration number 1 and 2 respectively.
The Date and Time fields are used to write down the date and time when the risk was identified.
Activation Date fields contain the date range when iteration is in progress.
Risk Description depicts the risk identified. Risk Removal Necessity Level is either in form of low or high. When it is very necessary
 to remove the risk for the project to succeed, its value is high, otherwise it is low.
No. of Tasks affected is the number of tasks during sprint or iteration that is affected by the identified risk. Usually, a user story from the product backlog is split into tasks in the sprint or iteration stage for a particular user story implementation by the developers before actual building of that module of the software takes place.
Task Description is a descriptive name for a task.
Task Relevancy Level takes the value of high or low. If the affected task is a very important one for the software project to succeed then relevance level is high or else it is low.
Planned Hour: This is the actual number of hour stipulated to execute a particular task and it is in minutes.
Overtime is the additional time to implement a rework of a faulted task.
Involved Person is the person charged with the execution of a particular task. Typically, this is a development team member who is usually a developer.
Rank: This consists of the various developer ranks in terms of experience in the software development. In this project, high-skilled person (hs), skilled person (s) and semi-skilled person (ss) are the ranks used. Total is the sum of rework hours for all tasks relating to the identified risk. Task Number uniquely identifies a task relating to the identified risk.

4. MATERIALS AND METHODS

This study utilized a hybrid of Object-Oriented Analysis and Design and Agile Software Methodologies.
UML Use cases and Use Case models of Object-Oriented Analysis and Design Methodology was developed. It is used in the study to describe the various activities performed and in the order they are performed and the actors that performed them. The activities include requirements to be met in the model built. The

requirements in this case are how the various budgetary activities takes place. The Use Case Model of the Extended Agile Software Development Project Model is shown in Figure 6. Agile methodology is also utilised in the steps followed to create iterations and make necessary updates of budgetary activities during iterations. This enables budgetary updates of various expenses that made up the risk management cost during iterations.
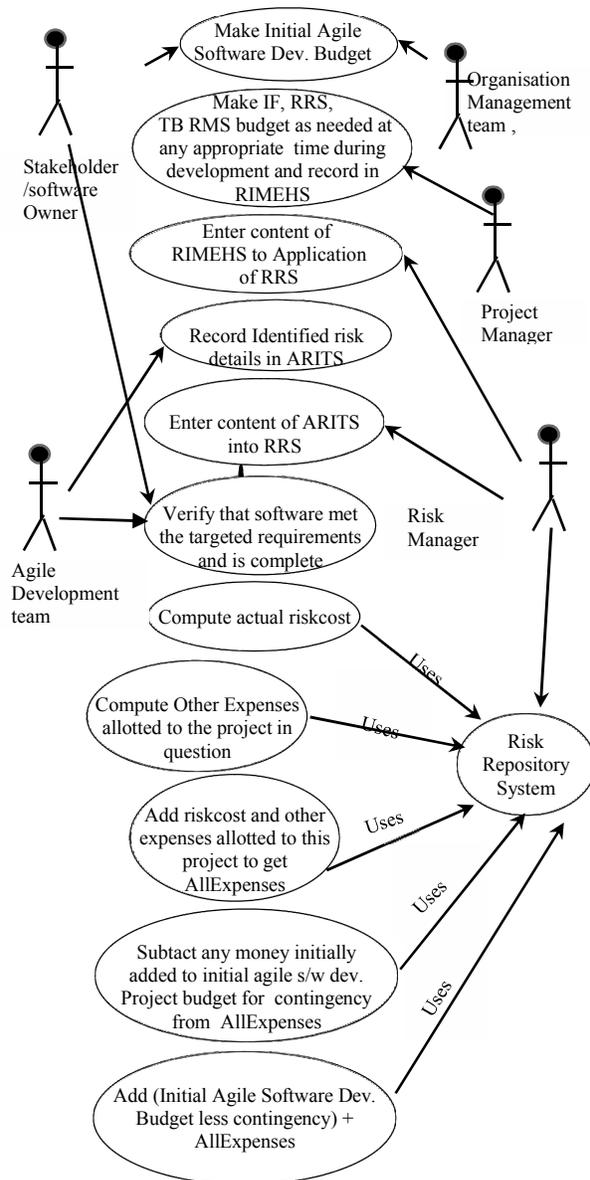

Fig. 4. Use Case Model for Agile Software Project Budget Model

5. EXTENDED AGILE SOFTWARE DEVELOPMENT PROJECT BUDGET STEPS

The Extended Agile Software Development Project Budget Steps include the following:

1. Initial Agile Budget Analysis
2. Risk management Budget Analysis /Evaluation Process
3. Risk Management Budget Computation
4. Final Agile Software Project Budget Arrival

1. Agile Budget Analysis

   i. Initial Agile Budget Analysis
       This is the first step during an agile analysis stage when business analyst, expert users and and agile team representatives meet to discuss and understand the functions of the software to be built and what it will likely cost to develop it. Frequently, some allowances for contingencies are slightly made; however these are not discrete from the software budget cost since explicit management of risks is not considered.

   ii. Risk Management Budget Analysis
       This is the second step in the Agile analysis stage where the contingency earlier mapped out in the initial agile budget analysis stage is separated from the initial agile software development budget, re-evaluated and adjusted to incorporate explicit management budget. This is achieved by assigning a percentage to risk management. From previous works, 15% of the development money is assigned for this purpose. However, this is not general as it depends on criteria used to evaluate risk management budget by organizations, size and complexity of the software developed, knowledge from experiences gotten from previous software project development on monies expended as a result of risks occurrences.

   iii. Risk management Budget Evaluation process
       This takes place from the commencement of iterations' − during iteration, planning and iteration, until software is verified to be complete. In other words, as risks are identified, assessed, monitored and reworked, monies are expended to prevent their occurrence and/or rework the risk conditions (tasks) associated with them as necessary risk management budget evaluation is built up. Likewise , when other risk management related expenses, namely

infrastructural expenses, training expenses, risk repository expenses and risk manager remuneration expenses are documented on the RIMEHS sheet by the project manager as their need surface and handed over to the risk manager who enters them into the risk repository via the repository system for updates and computation, risk management budget evaluation is further built up and updated. Also, the risks identified are in like manner documented in the ARITS sheet by the respective agile team members and the repository system updated by the Risk manager. These activities are simultaneously done with iteration activities but are the responsibilities of the dedicated risk manager to prevent compromising the swiftness of the agile process. It is worthy to note that apart from expenses emanating from managing risks and rework of tasks affected, other expenses are computed such that only a portion allotted to the given project is added to its risk management cost. This depends on the size and complexity of the software project as well as the stipulated period of time for which the share formula is made. For instance, training expenses for a year might be divided amongst the project executed in that year.

iv. Overall Risk Management Budget Computation
At this stage, it is verified that the software is complete, i.e had met its target requirements and ready for full release. The cost of risk and other expenses is computed using the repository system

v. Final Agile Software Project Budget
At this stage, the Overall Risk Management Budget obtained is added to the Agile software development project budget arrived at after deduction of contingency budget from the initial agile budget in stage two of this model. The result of this is the Extended Agile Software Development Project Budget when explicit risk management is incorporated into the agile process. The Extended Agile Software Development Project Budget Model is depicted in figure 5.

6. CONCLUSION

An Extended Agile Software Development Budget Model has been developed. It is a model that is built to incorporate budgetary consideration of explicit risk management with that of a normal Agile software project budget. This is necessary since the importance of explicitly managing risk in agile projects is increasingly being stressed as seen in the literature researched. However, the way to include cost that will accrue in implementing it is not known yet. In this paper, the constituent expenses to be made if explicit management will be included in the agile project was first derived, and thereafter a model that has the steps to achieve its inclusion in the budget of Initial Agile project budget was developed. This will enable stakeholders to know the various expenses that will emanate in the agile software development projects for explicit risk management to be incorporated. This is important as software development using agile had been limited to small software projects using the inbuilt implicit approach of risk management in agile method. Knowing the things involved is a first step to finding the right formula to estimate its cost. The detailed estimated mathematical formula will be presented in a future study which is already ongoing. That notwithstanding, the model developed though theoretical gives insight into how explicit risk management expenses will be added to agile software development budget.
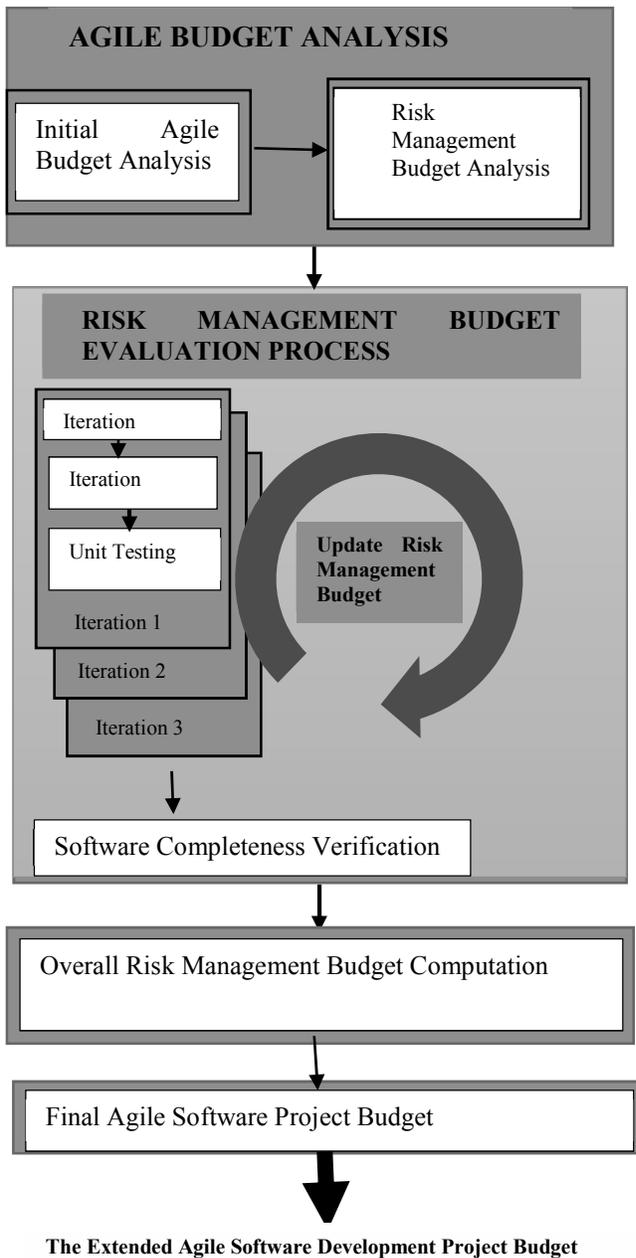
The Extended Agile Software Development Project Budget

Fig. 6. Extended Agile Software Project   Budget Model

REFERENCES

[1] Alharbi, E. T., Rizwan, .M. & Qureshi, J. (2014). Implementation of Risk Management with SCRUM to Achieve CMMI Requirements. Computer Network and Information   Security,11,20-25. DOI:10.5815/ijcnis.2014

[2] Agile Alliance, A. (2015). Agile 101. Retrieved from https://www.agilealliance.org/agile101/

[3] Caseley, S. (2014). Being agile within an existing delivery environment. Paper presented at PMI® Global Congress   2014—North America, Phoenix, AZ. Newtown Square, PA: Project Management Institute.

[4] Cohn, M. (2014). Succeeding with agile: software development using Scrum.   Pearson Education.

[5] Collier, P.M., Berry, A.J., Burke, G.T. (2007), Risk and   ManagementAccounting: Best Practice Guidelines for Enterprise-Wide Internal Control Procedures. Oxford: Elsevier

[6] Deloitte Development LLC. (2017). Retrieved from Bersin   by   Delloite: http://www.bersin.com/Lexicon/Details.aspx?id=150 22

[7] Dhlamini, J. N. (2009). Intelligent risk management tools for software development. Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers' Association: SACLA '09 (pp 33-40). New York, NY, USA: ACM . Retrieved from http://dx.doi.org/10. 1145/1562741.1562745

[8] Highsmith, J. & Cockburn,   A. (2001). Agile software development: The people factor. IEEE Computer 34(131-133).

[9] Islam, S. (2010). Software Development Risk Management Model- a goal-driven approach (Doctoral   Dissertation).   Retrieved   from http://mediatum.ub.tum.de/doc/1002328/1002328.pdf

[10] Khatavakhotan, A. S. (2011). MRMM: A Mathematical Risk Management Model for Iterative IT Projects based on the Smart Database. International Journal of Information and Electronics Engineering, 1(3), 229-233.

[11] Kontio, J. (2001). Software Engineering Risk Management: A Method, Improvement Framework, and Empirical Evaluation (Doctoral Dissertation). Retrieved   from   http://lib.tkk.fi/Diss /2001/isbn951225   655X/isbn951225655X.pdf

[12] Madden, D. (2014, December 29). Your Agile Project   Needs a Budget, Not an Estimate [blog post]. Retrieved]from https://hbr.org/2014/12/your-agile-project-needs-a-budget-not-an-estimate

[13] Mathkour, H. Assassa, G. & Baihan, A. (2008). A Risk Management Tool for Extreme Programming. International journal of Computer Science and Network Security 8 (8)

[14] Nyfjord J. (2008). Towards Integrating Agile Development And Risk Management Stockholm University , Department of Computer   and Systems Sciences. Sweden: Universite tsservice US-AB, Kista.   Retrieved   from https://www.divaportal.org/smash/get /diva2:199663 /FULLTEXT01.pdf

[15] Nyfjord J, & Kajko-Mattsson M (2008) Integrating Risk Management with Software Development: State of Practice. In: Proceedings of the International Conference on Engineers and Computer Scientists, Vol. I, pp 878–884

[16] Veehil, S. T. (2013). Risk Management in Agile. Retrieved   from   Scrum   Alliance,   Inc:

https://www.scrum alliance.org/community/articles/2013/ 2013-may/risk-management-in-agile

[17] Wiegers, E. K (2007). Practical Project Initiation: A hand book with Tools. Know your enemy. 77-88 Microsoft Press. ISBN 978-07356-2521-1. Retrieved from www.projectinitiation. com/book info/index.html: Downloaded 11/11/208