Journal of Theoretical and Applied Information Technology
10th October 2014. Vol. 68 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SOFTWARE RELEASE PLANNING- A MODEL INCORPORATING ENVIRONMENTAL PARAMETERS

[1]SANDHIA VALSALA, [2]DR. ANIL R NAIR
[1]Ph.D Scholar, Karpagam University, Coimbatore
[2]Principal Scientist, ABB Corporate Research, Bangalore
E-mail: [1] sandhia_v@yahoo.com, [2] aniltsp@yahoo.com

ABSTRACT

A software release planning can be seen from two dimensions "what to release" and "when to release". The most crucial decision is whether or not to select a feature for implementation in the next software release. A number of software release planning models are available which considers a wide variety of factors in deciding the implementation of a feature in a release.This paper analyzes 31 release planning models and the selection factors used by these models. Most of these models use only in-project parameters in deciding on the features to be included in a release. A new release planning model incorporating a group of "environmental factors" ,which plays a crucial role in deciding the priority of features to be included in each release is then proposed .The paper emphasize the need to include environmental parameters which are parameters not directly linked to project, but influences the project from outside in planning a release.

Keywords: *Software Release, Release Planning, Environmental Parameters, In-Project Parameters, Feature Priority*

1. INTRODUCTION

Release planning is a problem of deciding on the features that has to be included in subsequent releases. This decision is dependent on various technological and resource constraints.[1,2] The objective of planning is to find the best composition of features to be included in a release .A variety of methods and techniques do exist in formulating a release planning problem. A poor release planning decision can result in a release lacking customer satisfaction, quality, not offering the best business value and not meeting the needed constraints. A good release plan addresses all decisions related to the selection and assignment of features to a sequence of consecutive product releases. [14].This paper does a systematic review and analysis of 31 most popular release planning models with a proper grouping of selection factors used by these models during release planning. After analyzing these selection factors the most crucial in-project parameters are identified. A new theoretical framework for release planning is developed, incorporating "environmental parameters" which can also play a crucial role in deciding the priority of features to be included in each release.

2. RELEASE PLANNING MODELS

The following systematic planning models were considered in this study.

Cost Value Approach (CVA)
The model [6] focus on prioritizing software requirements based on stakeholders preference. According to J.Karlsson and K.Ryan a software system can succeed only if its quality is maximized, cost is minimized and it's delivered fast. CVA model prioritize requirements based on their relative value and cost as prioritization based on relative rather than absolute assignments as it is faster more accurate and most trustworthy according to the authors. This model is not used in industry, but is validated using two case studies. [4]

The Incremental Funding Method (IFM)
This model [7] uses a data driven financially informed approach to software development by analyzing and sequencing feature delivery by maximizing the Net Present value (NPV).

Evolve
The model [8] provides an evolutionary and iterative approach which offers decision support for release planning. The model provides optimum allocation of requirements to releases, determines stakeholder conflict, and balances the resources to all the releases. This solution approach is supported

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

by a tool Risk-optimizer. The model is validated by a case-example on a sample project. [4]

Evolve+
The model [9] is an extension of Evolve and is a combination of computational genetic algorithm and iterative method. The model is developed based on industrial feedback, and also considers effort and risk associated with requirements. The model finds the most suitable solutions from the list of available solutions. The model is validated on a sample project in academia and on two industrial case studies. [4]

Evolve*
The problem of deciding which requirements should be assigned to which release is discussed and this proposed hybrid approach called EVOLVE* model [10] improves existing methods for release planning by combining the strength of mathematical models (complexity, size) with experts' knowledge. It is designed for two releases in advance and is validated by two case studies. [4].

S-Evolve *
It is an approach to solve release planning problems for evolving systems. The feature to be included in the new system arises from various stakeholders' preference, despite the available resource and risk constraints. The model [11] considers knowledge about existing software product as the core to making meaningful release decisions. The functionality and characteristics' of existing system is also considered by this model. The model is validated through a case study performed on a real system. [4]

F- Evolve *
F-EVOLVE* model [12] may be used to decide which features to produce and when based on their financial contributions. Specifically, F-EVOLVE* may be used to determine which features generate the highest returns, with the shortest development time. The model is validated on a web portal project of Epcour. [4]

Evolve ^{ext}
The model [13] is an extension of EVOLVE *.The model addresses the assignment of requirements to releases on a strategic level. The factors considered are effort, finance and risk constraints. The goal is to find an optimal balance between competing stakeholder priorities and bottleneck resources. It combines computational and human intelligence to solve the wicked problems of release planning. The model is validated on an Industrial case study. [4]

Art and Science of Release planning model (AHPSRP)
A hybrid release model [14] that integrates the strength of computational intelligence and knowledge and experience of human experts in feature prioritization. The model uses human intuition to formalize the problem and applies computational algorithm to generate the best solutions. The model was validated on a sample project involving 15 features and two stakeholders. [4]

Evolutionary EVOLVE+
Evolutionary Evolve+ [15] is an extension of hybrid intelligence approach EVOLVE*. This approach adds soft constraints and objective of RP to decision making process that were ignored in all previous approaches. Due to the cognitive and computational complexity of problem, optimization (computational complexity) and multi-criteria decision (cognitive complexity) are combined to formulate new approach EVOLVE+. The model is validated on a real world case study. [4]

Next Release Problem (NRP)
The model [16] uses heuristics to solve the problem of release planning..NRP uses exact optimization technique .The model considers the following feature selection parameters like customer's value, requirement cost and number of basic requirement of customer.

Multi Objective Release Planning (MORP)
This model is closely related to SBSE (Search based Software Engineering) and it uses multi objective optimization technique. [17]

Multi Objective Next Release Planning (MONRP)
MONRP is a model [18] in which customers with varying requirements are targeted for the next release of existing software. Selection of a requirement involves spending some resources which can be converted to cost and also to provide value to the company. The problem is to select the set of requirements that maximize total value and minimize the required cost in order to optimize both value and cost simultaneously. It considers each objective independently in order to explore search space towards parento-optimal front. In the formulation of MONRP two objectives are taken

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

into consideration Maximize customer satisfaction and minimize required cost. The following search techniques are used NSGA-II (Non dominated Sorting genetic algorithm), Parento GA, Single objective GA and Random Search.

Bi-Objective Release planning for evolving systems [BORPES]
Most of the existing Release planning methods do not consider the existing system in making release RP decisions. This model [19] detects the coupling between features based on relatedness of the components that would implement the feature. This model includes highly coupled features in the same release by considering both feature coupling in solution domain and problem domain. The model was validated by a case study based on the available data from Release Planner. [4]

An Evolutionary Quantitative Win Win Approach (AEQWW)
The proposed method [20] called Quantitative Win Win uses an evolutionary approach to provide support for requirements negotiations. This model combines quantitative models with iterative approach to determine the best requirements. The model is validated by a small scale example using GENSIM simulation model. [4]

Analytical Model for requirements selection Quality Evaluation [AMRSQE]
Here an analytical model [21] of the selection process is presented which takes the quality of the decision-making into account. The model is a network of queues with multiclass jobs corresponding to requirements of different quality. The analytical model can act as a baseline for simulation of more realistic models where no analytical solution is possible. Two surveys were conducted to validate the feasibility of this model. [4]

Quality Performance Model (QUPER*)*
This model [22] is used in Industry and is developed on the basis of existing method "*cost-value* approach" .QUPER develop release plans on the basis of quality requirements, as existing approaches not consider quality aspect at this level for release planning. The model is partially introduced at Sony Ericcson. [4]

A Mathematical Formalization for Flexible Release Planning (AMFFRP)
This model [23] uses Integer Linear Programming (ILP) by introducing a unique set of aspects (constraints considered during release planning) and managerial steering mechanisms. The model is validated by an industrial case study. [4]

Release Planning with Feature Trees (RPFT)
The model [24] describes how to utilize feature trees for planning the releases of an evolving software solution and evaluates the effects of the approach on effort, decision-making, and trust. The model is validated on an industrial case study.

MAX-MIN Ant System with a Dynamic RouletteWheel (MMASDRW-SRP)
The model [25] adopts a heuristic approach based on **ant colony optimization** (ACO) and can be applied to obtain satisfactory suboptimal solutions within a reasonable amount of computational cost. The study uses the problem instances based on the PSPLIB database of the multi-mode resource constrained project scheduling problem (MRCPSP).

Release Plan Simulator (REPSIM-1).
The model [26] combines computational method with human expertise to formulate and analysis solution. Motivation to develop this approach is uncertainties in different factors which impact the RP decisions. Features assigned to release may change over time, so it is very important to make sure that to what extent proposed release plan remain stable. The model is validated on a case study in academia. [4].

RP with Fuzzy Effort Constraints (RPUFEC)
The model [27] aims in finding an appropriate release plan to maximize stakeholder's satisfaction. In this method two fundamental paradigms uncertainty and intelligent decision support are combined. Fuzzy logic is used to handle uncertainty of data regarding effort estimation, effort constraints and objectives related to cost, benefit and quality. The model is validated by a case study example in academia. [4]

Quality Improvement Paradigm (QIP)
The model [28] introduces a six step process for release planning. The goal of this approach is to deliver maximum value to the customer in least time possible. It combines the computational strength of genetic algorithms with the flexibility of an iterative solution method. In QIP learning from previous release data is considered important and this previous knowledge can be useful for improvements in future releases. The model is validated by testing in a real world environment at iGrafx Corel Inc. [4]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

An Optimization technique for RP (AOTRP)
The model [29] uses Integer Linear Programming (ILP) techniques to help product and requirements managers in software release planning. The model optimizes revenues against available resources in a given time period. The model is validated by conducting an experiment on a scenario of a development organization. [4]

Fuzzy Model for dependence constraints in RP (FMDCRP)
The model [30] improves on existing methods for release planning by handling the uncertainty of data using fuzzy logic. The model uses fuzzy logic to model the uncertainty concerning the identification of structural dependency constraints between requirements. This model is developed to remove the uncertainties regarding requirement dependencies for RP.The model has been validated by a case example. [4]

Fuzzy Optimization Model for RP (FOMRP)
Release planning decisions are required at an early stage in the development cycle, when uncertainty is unavoidable in the project estimates. The model [31] uses fuzzy theory to address issues concerning the uncertainty in the release planning problem: fuzzy effort constraints and fuzzy dependency constraints.

Consensus-Driven and Value based RP approach (CDVBRPA)
It is an effective release planning and configuration method used in small organizations. The model [32] analyze, prioritize requirements, and finds a candidate release configuration that can be developed within the time, quality and functionality constraints relating to the project. The method uses value-based and consensus-driven approach in solving RP problems. The model is validated in an experiment conducted in academia. [4]

An Interactive and explanation oriented dialogue approach for RP

3. TAXONOMY OF SELECTION FACTORS - ANALYSIS AND INTERPRETATION.

A detailed taxonomy of selection features used by various software release planning models is shown in the table given below.

The dialogue approach [33] is aimed at reducing the complexity of problem during interaction with the human expert. It is used for planning of wicked and complex problems. The model is applied on real world problem and is not validated by a case study or experiment. [4]

Post Release analysis of requirements Selection Quality (PARSEQ)
PARSEQ method [3] is aimed to improve release planning decisions that are made in previous releases and it is based on retrospective analysis as a way to look back at the events taken place. Quality of selected requirements in a release and quality of RP process (requirement selection process) is analyzed for proposing improvements. For analyzing quality of selected requirements, the cost and value of each requirement is re-estimated and wrong selected requirements or incorrect decisions (about requirement selection) are inspected. This method is also useful for re-prioritization of requirements for future releases or re-prioritization of requirements in all sequence of releases. The model is validated through two industrial case studies [4].

Risk driven method for Extreme Programming (RDMXP_RP)
The model [34] is suitable for small teams, lightweight projects and vague and volatile requirements. It is a risk-driven method for XP release planning. The model is validated in industry on a web based application project. [4].

Hybrid approach Incorporating CP with RP (RP&CP)
The model [35] uses an hybrid approach combining the strengths of Constraint Programming (CP) and Release By (RP).It uses a two staged solution approach which combines the higher flexibility in problem formulation (in terms of describing objectives and constraints) of CP with the advantages offered by RP. The model is validated on a real world data set with 600features.

Journal of Theoretical and Applied Information Technology
10th October 2014. Vol. 68 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table1. Taxonomy of Feature Selection factors

Software release Planning Model	Selection Factors																	
	Stake holder Preference	Cost	Value	Effort	Time	Requirement/Feature Dependency	Resource	Risk	Budget	Revenue	System Constraint Product Feature	Deadlines	Business /Market values	Quality	Non Functional Requirement	Urgency	Fuzzy Constraints	Development Team
CVA	√	√	√															
IFM		√		√	√													
EVOLVE	√			√		√												
EVOLVE +	√			√		√	√	√										
EVOLVE*	√			√		√	√		√									
F-EVOLVE		√			√	√	√			√								
EVOLVE EXT	√				√	√												
S-EVOLVE	√					√	√				√							
NRP	√	√				√			√									
AHPSRP	√			√	√				√									
MORP	√	√				√	√	√				√						
MONRP	√						√											
BORPES	√					√							√					
AEQWW	√			√	√									√				
AMRSQE				√		√			√				√					
REPSIM				√			√											
QUPER		√													√			
AMFFRP						√	√					√						√
RDMXP-RP		√	√	√		√							√					
PARSEQ	√	√	√			√	√											
MMASDR W	√					√	√									√		
RPUFEC				√			√										√	
QIP				√		√	√											
AOTRP						√	√			√								
FMDRCP																		√
CDVBRPA					√	√										√		
FOMRP				√		√	√											
DIALOGUE APPROACH IN RP				√		√												
RPFT	√			√	√						√							
RP&CP	√					√	√								√			

All these found models provide different solutions of strategic RP and discuss different requirements selection factors. Most of the models discussed above donot categorize these selection factors but rather gives only the description and use of these factors in their model. There are many common requirements selections factors among the majority of identified models. It is observed that almost 70% of models consider technical constraints (requirements dependency and others) during

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

planning strategic release. Similarly, 50% of found models emphasize on resource constraints (available resources and required resources) and effort constraints (required effort) for road mapping. The stakeholders" influence in requirements selection is highlighted by 53 % of the models. Only two models *QUPER* and RP&CP discuss strategic RP from non-functional requirements perspective and underline the need of selecting requirements on the basis of desired quality attributes required in a release. Similarly, there are two models *S-Evolve* and RPFT* that discuss system constraints for selecting requirements in a release on the basis of already delivered system or release. System constraints are related to modification of already developed requirements during the development of a new release.

4. ENVIRONMENTAL PARAMETERS

Environmental factors can be defined as parameters that can influence a project from outside and are important as in- project parameters in software performance prediction and software development. A proper identification of these environmental factors is essential. It has already been proved that efficient use of environmental parameters have made software performance prediction more robust. Hoang Pham[36] has identified a set of environmental parameters to be used in software reliability. Dr Anil [36] has suggested five environmental parameters that could be considered in software performance prediction and it has been proved that the inclusion of environmental parameters highly improved the performance predication of the software. The five environmental factors identified for a better software performance prediction were Group maturity rating, defect rating, Project risk index, project Compliance Index, and coefficient of variation of historical data. None of release planning models consider working environment in planning a release. All the models assume a constant environment while developing software and this can have a negative impact on the developed software. Incorporating environmental parameters in software release planning will definitely give a better feature prioritization results so that the product will be beneficial both to the customer as well as to the organization. An attempt has been done here to identify some environmental parameters that could be used in software release planning. These identified environmental factors are explained below and it can be used along with any traditional release planning model.

5. PROPOSED RELEASE PLANNING MODEL

Majority of the release planning models uses only in project parameters in feature selection but the proposed release planning model uses two types of parameters for feature selection ,Model (in-project) parameters and Environmental parameters (organizational view).

Model Parameters

31 most popular release planning models have been analyzed and the most critical factors for feature selection were identified as Stakeholder preference, Requirement dependency and Resource constraints. **Stake Holder Preference** is the priority value provided by the stakeholders who are those people or organization who will be affected by the system and who have direct or indirect influence on system requirements. **Requirement Dependency** can be defined as relationships between two or more requirements in terms of implementation. Precedence and coupling are example of technical dependencies **Precedence** is a relationship, when one requirement cannot be implemented before other requirement and **Coupling** is a relationship, when two requirements are to be implemented together in a release. **Resource constraints** can include resource restriction or limitation and includes various resource constraints like budget, schedule, risk and effort

Environmental Parameters

Environmental parameters are those parameters which are not directly linked to project, but which influences the project from outside. It is more from organizational point of view. The following environmental factors were identified as Historic data, Availability and Productivity of developers, Uncertainty, Financial Stability of the company vs Resources constraints and Competitive advantage. **Historic Data** is the data collected from similar projects developed previously. Various parameters can be measured from the historic data like defect density, effort estimation, customer satisfaction and project duration variance. These parameters when measured from historic data can be a crucial deciding factor in feature selection. **Availability and Productivity of Developers** as the proposed model is to be tested in the planning and development phase of a release this parameter will have a major influence on feature prioritization. One must never forget that the output of the developed software lies in the hand of these developers. **Uncertainty** is an unavoidable issue

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and is related to operational release. This has a major impact on feature selection and has to be included as a critical factor in feature selection. **Financial stability vs Resource constraints** can be another deciding parameter in feature prioritization. The estimation of cost and resources needed in implementing a feature is considered as a Model parameter but the selection of a feature cannot be only based on these resource constraints. These estimated values have to be checked with the financial aspect of the company and a final decision on the inclusion of the feature has to be decided. **Competitive advantage** can be another crucial deciding factor for feature selection. If any competitor has already implemented this feature and has got a negative response definitely the feature has to be removed from our selection list.

In addition to that most of the Software release planning models is used for planning a release (Planning Phase). The features which are identified in the planning phase are developed and implemented in the next release. The model parameters do not vary with environment .The proposed model in addition to these in-project parameters also uses environmental parameters in feature selection. Since these environmental parameters may vary with environment and is more from organizational point of view, the proposed model can be applied during the planning phase and development phase of a release. Rather than just using the in- project parameters in feature selection the use of environmental parameters provides a better selection of the features to be included in a release.

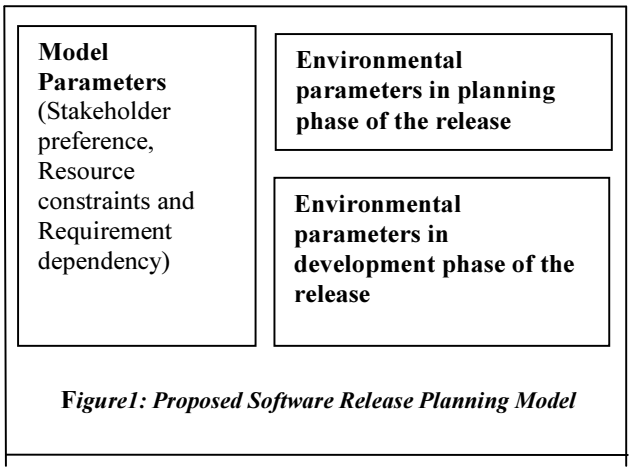

Figure1: Proposed Software Release Planning Model

6. CONCLUSION

The paper introduces five environmental factors that can be crucial in prioritizing features to be included in a release. Most of the existing software release planning models use only in project parameters in feature selection. Using environmental parameters which are parameters from organizational point of view and that are not directly linked to the project can improve release planning decisions. The identified environmental parameters are Historic data, Availability and Productivity of developers, Uncertainty, Financial Stability of the company vs Resources constraints and Competitive advantage.

As a future work the actual implementation of the proposed release planning model incorporating the environment parameters need to be done on real world data to analyze the impact of these parameters on feature prioritization.

REFERENCES

[1]http://en.wikipedia.org/wiki/Maintenance_release
[2]http://www.spamlaws.com/windows-operating-system.html
[3] Hans Christian Benestand, Jo E Hannay "A Comparison of Model-based and Judgement based Release planning on Incremental Software Projects"
[4] Saad Bin Saleem and Muhammad Usman Shafique , "A Study on Strategic Release Planning Models of Academia and Industry Through Systematic Review and Industrial Interviews", School of Engineering Blekinge Institute of Technology ,Sweden
[5] Mikael Svahnberg_, Tony Gorschek, Robert Feldt, Richard Torkar,Saad Bin Saleem, Muhammad Usman Shafique,"A Systematic Review on Strategic Release Planning Models", Blekinge Institute of Technology,Sweden
[6] J. Karlsson and K. Ryan."A cost-value approach study for prioritizing requirements", IEEE software, 14(5):67–74, 1997.
[7] Denne, M. and Cleland-Huang, J., "The Incremental Funding Method: Data Driven Software Development," IEEE: pp. 39-47, 2004.
[8] D.Greer and G. Ruhe."Software release planning: an evolutionary and iterative approach". Information and Software Technology, 46(4):243–253, 2004

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[9].Ruhe, G., and Des, G., "Quantitative Studies in Software Release Planning under Risk and Resource Constraints", Proceedings of the 2003 International Symposium on Empirical Software Engineering, pp. 262-270, 2003.

[10] Ruhe, G., and Ngo, A., "Hybrid Intelligence in Software Release Planning", Int. J. Hybrid Intell. Syst. vol.1, no.1-2, pp.99-110, 2004.

[11] Saliu, O., Ruhe, G., "Supporting Software Release Planning Decisions for Evolving Systems", 29th Annual IEEE/NASA Software Engineering Workshop, pp. 14-26, 2005.

[12] Maurice, S., Ruhe, G., Saliu, O., and Ngo-" A., "Decision Support for Value-Based Software Release Planning", Journal of Value-Based Software Engineering, pp. 247-261, 2006

[13] Ruhe, G., Momoh, J., "Strategic Release Planning and Evaluation of Operational Feasibility, "Proceedings of the 38th Annual Hawaii International Conference on System Sciences, HICSS,,05, pp. 313b-313b, 2005.

[14] G. Ruhe and M. O. Saliu. "The art andscience of software release planning". IEEESoftware, 22(6):47–53, 2005.

[15] Ngo-The, A., and Ruhe, G., "A Systematic Approach for Solving the Wicked Problem of Software Release Planning", Soft Comput, vol. 12, no.1, pp. 95-108, 2007.

[16] Fabricio G Freitas, Daniel P Coutinho, Jefferson T Souza, "Software next release planning approach through exact optimization," International Journal of Computer Applications, May 2011.

[17] "A New Approach to the Software Release Planning", Brazilian Symposium on Software Engineering, 2009

[18] Yuanyuan Zhang, Mark Harman,S Afshin Mansouri ,"The Multiobjective next release problem", Proceedings of the 9th annual conference on Genetic and evolutionary computation

[19] Saliu, O., and Ruhe, G., "Bi-objective Release Planning for Evolving Software Systems", In Proceedings of the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering (ESEC-FSE '07), pp.105-114, 2007.

[20] Ruhe, G., Eberlein, A., and Pfahl, D., "Trade-off Analysis for Requirements Selection", International Journal of Software Engineering and Knowledge Engineering, vol. 13, pp. 345-66, 2003

[21] Regnell, B., Karlsson, L., and Martin, H., "An Analytical Model for Requirements Selection Quality Evaluation in Product Software Development", Proceedings of the 11th IEEE International Conference on Requirements Engineering, pp. 254-263, 2003.

[22] Regnell, B., Svensson, R.B., and Olsson, T.,"Supporting Road mapping of Quality Requirements", IEEE Software, vol.25, no.2, pp.42-47, 2008

[23] Van den Akker, M., Brinkkemper, S., Diepen, G. and Versendaal, J., "Software Product Release Planning through Optimization and What-if Analysis," Information and Software Technology, vol.50, issue.12, pp.101-11, 2008

[24] Samuel Fricker1 and Susanne Schumacher," Release Planning with Feature Trees: Industrial Case", REFSQ 2012, LNCS 7195, pp. 288–305, 2012, Springer-Verlag Berlin Heidelberg 2012

[25] Qing Huang Yu, Wen Chiang Chuan, Hsu Huang Cheng, "Applying a MAX-MIN Ant System with a Dynamic Roulette Wheel Strategy to Software Release Planning", International Conference on advance computer science and electronics Information, July 2013

[26] Pfahl, D., Al-Emran, A., and Ruhe, G., "A System Dynamics Simulation Model for Analyzing the Stability of Software Release Plans", Software Process Improvement and Practice, vol.12, pp. 475-490, 2007

[27] Ngo, A., Ruhe, G., and Wei, S., "Release Planning under Fuzzy Effort Constraints", Proceedings of the Third IEEE International Conference on Cognitive Informatics, pp. 168-175, 2004

[28] Amandeep, A., Ruhe, G., and Stanford, M., "Intelligent Support for Software Release Planning", 5th International Conference on Product Focused Software Process Improvement, pp. 248-262, 2004

[29] Van den Akker, J. M., Brinkkemper, S., Diepen, G., and Versendaal, J., "Determination of the Next Release of a Software Product: An Approach using Integer Linear Programming", Proceeding of the Eleventh International Workshop on Requirements Engineering, vol.10, pp. 119-24, 2005.

[30] Ngo-The, A., and Saliu, O., "Fuzzy Structural Dependency Constraints in Software Release Planning", The 14th IEEE International Conference on Fuzzy Systems (FUZZ '05), pp.442-447, 2005

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[31] Ngo-The, A., and Saliu, O., "Measuring Dependency Constraint Satisfaction in Software Release Planning using Dissimilarity of Fuzzy Graphs", Fourth IEEE Conference on Cognitive Informatics, (ICCI 2005), pp. 301-307, 2005

[32] Ziemer, S., Falcone Sampaio, R. P., and Stalhane, T., "A decision modeling approach for analyzing requirements configuration trade-offs in time-constrained Web Application Development", Eighteenth International Conference on Software Engineering & Knowledge Engineering (SEKE'2006), pp.144-149, 2006.

[33] Du, G., Richter, M. M., and Ruhe, G., "An Explanation Oriented Dialogue Approach and Its Application to Wicked Planning Problems," Journal of Computing and Informatics, vol. 25, pp. 223-49, 2006.

[34] Mingshu, L., Meng, H., Fengdi, S., and Juan, L., "A Risk-Driven Method for Extreme Programming Release Planning", Proceedings of the 28th International Conference on Software Engineering, pp., 2006.

[35] Mark Przepiora, Reza Karimpour, Guenther Ruhe "A Hybrid Release Planning Method and its Empirical Justification", 2013.

[36] Dr Anil R Nair, "Effective Performance prediction using environmental parameters", PhD Thesis, IT Mumbai,2008.