International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-3 Issue-5, June 2014

Review and Analysis of Software Release Planning Models

Sandhia Valsala, Anil R

Abstract: ***Software maintenance is generally recognized to consume majority of resources in many organizations. Regular replacement of legacy systems with new ones is not a feasible solution. Planning releases so as to maximize the functionality and quality of software is very much in need now. Release Planning plays a very important role in managing and maintaining releases and helps in the delivery of a high quality product to the end-users. Software Release planning involves proper grouping of activities in the release of one or more versions of software to one or more customers. This paper analyzes the various release planning models and the factors considered by these models for feature selection. 32 release planning models are considered and taxonomy of requirement selection factors is constructed. The main contribution of this paper is to assist software engineers in finding out the real factors that need to be considered in planning a release and to assess the effect of these identified factors on a release so as to plan releases efficiently and effectively.***

IndexTerms- Release planning, Software maintenance, Legacy systems, Requirement selection.

I. INTRODUCTION

Release planning plays a very important role in managing successful releases to the customer. Release planning is a software engineering process which is very loosely and poorly understood in today's world and it is intended to oversee the development, testing, deployment and support of software releases[1]. Good Release planning practices ensure that when your software is built, it will be successfully delivered to the people who want to use it. Planning releases so as to maximize the functionality and quality of the software is very much in need now and it is the most time-consuming area of Release planning.

In simple terms Release planning can be described as a collection of new functionality that can be added to an existing production environment. Each release includes a set of features satisfying certain constraints of the organization. Deciding on what features to include and what not is very crucial in planning a release. There are so many factors that are considered in the selection of features like cost, effort, resources, time and stakeholder's preference. Most of the models considered in this study use different technical and non technical factors of requirement selection.

Manuscript published on 30 June 2014.
***** Correspondence Author (s)
Sandhia Valsala ,Phd Computer Science Student,Karpagam University, Coimbatore, India.
Dr.Anil R, Principal Scientist, ABB Corporate Research Bangalore, India.

A comparatative analysis of existing models for release planning is done with a proper grouping of selection factors. The contribution of this article is to help the managers to know the different release planning models and what requirement selection factors are used by these models

II.RELEASE PLANNING METHODOLOGY

There are two important approaches of release planning. Judgment based Release planning approach and Model based approaches. Hybrid approach uses a combination of ad-hoc and systematic approaches.

A) Judgment based Release planning

In some companies release planning is not considered as a separate activity. It relies only on the judgment and decision of the project leader or the team involved in developing the project. Some companies still rely on these judgment based models as these models ignore what experts consider important and there is often mismatch in the inputs of the model and that of experts[5].The study done by Hans Christian Benestand and Jo E Hannay also reveals that the models focus only on selected part of a possibly large space of relevant planning factors[5] .A study by Ruhe and Mamoh reports that as requirements changes often the judgment based planning is inefficient and that after the introduction of a model based release planning the planning required significantly less effort and stakeholders were more satisfied[6].

B) Model based Release planning

The following systematic planning models were considered and analyzed.

1) Cost Value Approach (CVA)

CVA approach makes pair wise comparison of customers requirements based on Analytic Hierarchy Process (AHP) according to their relative cost and value.AHP's pair wise comparison method to assess the relative value of the candidate requirements is done and also uses AHP's pair wise comparison to estimate the relative cost of implementing each candidate requirement and plots these on a cost–value diagram. After analyzing the cost– value diagram, software managers prioritize the requirements and decide which will actually be implemented.[6]

Factors Considered: Stake holders preference, Cost, Value

2) The Incremental Funding Method (IFM)

IFM decomposes project into MMF(Minimum Marketable Features) which is small self contained features that can be delivered quickly as such decomposition reduces risks and contributes to the success of the project. [7]

Factors considered: MMF's revenue value is measured in terms of tangible and non tangible factors like revenue generation, cost savings, competitive differentiation, brand name projection and customer loyalty. Cost and Effort for developing the MMF and time needed for a project to reach self-funding status is also considered.

3) EVOLVE FAMILY

Since several stakeholders are involved in a project two kinds of evaluation can be incorporated, one is assigning a perceived value to each requirement based on its impact on the final product or prioritization based on urgency of the system. Effort of an increment is the sum of effort of individual requirements assigned to this increment. Precedence and dependency constraints where in some requirements must be implemented before others. Given a set of requirements with their effort estimations and categorization of requirements into priorities by stakeholders and their technical constraints the method uses genetic algorithm to derive potential releases.[8]

Factors considered: Requirement dependencies, Stakeholder priority, Penalty, Benefit and Effort

4) EVOLVE +

It combines the strength of genetic algorithms with the flexibility of an iterative solution method. For each solution generated the constraints are checked like the effort constraint is handled by greedy like increment allocation problem .The risk constraint is checked by summing the total risk for each release and comparing it with risk referent. Precedence constraint, coupling constraint and resource constraint are implemented by specific rules and is used to check for an optimal solution. The solution is rejected if it violates the constraints.[9]

Factors Considered: Stakeholder preference, Effort estimate, precedence constraint, coupling requirements, resource constraints, and risk

*5) EVOLVE**

Solutions generated by EVOLVE* are optimal or near optimal. EVOLVE* consists of three main phases called modelling, exploration, and consolidation. Different from former algorithms of the EVOLVE family, the new approach plans only two releases in advance, i.e., each requirement is assigned to one of the following three categories: "next release", "next but one release", "not yet assigned". EVOLVE* aims to achieve maximum stakeholder satisfaction. The iterative procedure allows intelligent search of most promising solutions under the competing criteria of time, benefit and quality as described by the "magic triangle".[10]

Factors Considered: Inherent precedence, coupling and resource constraints, stakeholder preference, effort constraints, and budget constraints

*6) S-Evolve **

S Evolve* concentrates on ten key technical and nontechnical aspects impacting release planning. This model considers the effect of existing system characteristics on release planning decisions. Initial realization of this framework focuses on historical defect data. This proposed approach extends the existing solution method called EVOLVE by (i) the proactive analysis of the risk involved in integrating new features into existing components of the system and (ii) identifying the importance of estimating the integration effort for each feature based on system characteristic. Component Modifiability assessment is added which checks the feasibility of modifying system components.[11]

Factors Considered: Stakeholders value, Stakeholders satisfaction, technological constraints, Resource consumption, resource constraints, and System constraints

*7) F- EVOLVE **

F-EVOLVE* is an extension of EVOLVE* to accommodate the financial value in the form of net present value estimates of proposed features. This extension enables us to perform financial value-based software release planning.[12].

Factors Considered: Resource capability constraint, time constraint, feature dependency constraint, cost, and annual revenue per requirement.

8) EVOLVE ext

The model describes the mechanisms to reduce the complexity of strategic and operational planning to a series of data and represent input from all stakeholders. The model also provides improved planning and re-planning in a dynamic business environment, including the ability to validate strategic plans against operational limitations. In this approach operational feasibility of a proposed release plan is evaluated to extend the capabilities of strategic release planning. Resources are important part of releases and it is very essential to consider all possible resource types for an increment/release. In this approach three type of feasibility problem are formulated to validate the feasibility of proposed strategic release plan for next immediate release with respect to tasks and available resources. [13]

Factors Considered: Requirement dependencies, Stake holder preference, Time, Requirements volatility

9) Art and Science of Release planning model (AHPSRP)

Two type of dependency between features is considered here a coupling relation and precedence relation. Resource constraint includes budget time and effort estimation. Stakeholder preference is another element that is considered in this model. Feature prioritization is done in terms of value and urgency. An objective function is generated that tries to bring together the different aspects explained above in a balanced way.This approach formulates a series of problems as variants of the original formal model. Solving these problems to generate a set of qualified alternative solutions is done and a human decision maker evaluates the solutions and selects the best one.[14]

Factors Considered : Stakeholder preference, Resources like time, budget and effort.

10) Evolutionary EVOLVE+

The model goes through three phases, modeling, exploration and consolidation. Modeling is formulation step. In exploration phase, mathematical optimization algorithm is applied to get solution set (release plans). In consolidation phase evaluation of model and solutions are performed. In this evolutionary problem solving approach, suitable solutions are generated by the interaction between human expert and results of computational algorithms based problem description.[15]

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Factors Considered: Risks of implementation, resources consumption, stakeholder satisfaction and competitiveness, coupling, precedence, resource and budget constraints.

11) Next Release Problem (NRP)

Two mono objective metahueristics is used in solving the problem, Simulated Annealing and Genetic Algorithms. In **Simulated Annealing** if a new solution is better than the current solution then it is accepted otherwise if the new solution worsens the objective then it is accepted with certain probability defined in terms of difference between solutions. In **Genetic Algorithms** the metahueristic is defined in terms of two generic operations, crossover (structural information of two solutions are crossed to generate two new solutions) and mutation avoids the generation of same solutions thereby searching various search spaces. Exact techniques are methods that use mathematical operations to solve problems. Also an exact optimization technique using **revised Simplex** in the product form of inverse was also used. For solving Integer programming the method used is **branch and bound**.[16]

Factors Considered: Customers, Requirements (cost, requirement dependency) , company budget

12) Multi Objective Release Planning(MORP)

Each and every requirement that need to be considered is associated with a risk factor**.** Stakeholder preference and Resources are two other factors considered in this model. In this model the following relations are taken into account Stakeholder vs Requirements , Resources vs Releases vs Requirements , Risks vs Releases. The problem formulation of this model is as follows: maximize stakeholder satisfaction and minimize project risks respecting the available resources and requirements interdependencies.[17]

Factors Considered: Stakeholders preference, cost, deadlines, resources, risk, and requirement interdependencies.

13)Multi Objective Next Release Planning(MONRP)

MONRP is a model in which customers with varying requirements are targeted for the next release of existing software. Selection of a requirement involves spending of some resources which can be converted to cost and it also provides value to the company. The problem is to select set of requirements that maximize total value and minimize 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 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.[18]

Factors Considered : Customer preference, Resources

10)Bi-Objective Release Planning for evolving systems [BORPES]

Impact analysis is a process involved in identifying the elements of an existing system that will be affected by a change**.** A coupling between features that are yet to be implemented is to be done in order to sequence their implementation.[19]

Factors Considered: Features selection based on SD-coupling, feature value based on business perspective stakeholder value

15) An Evolutionary Quantitative Win Win Approach (AEQWW)

Firstly, it iteratively uses the Analytical Hierarchy Process (AHP) for a stepwise analysis with the aim to balance the stakeholders' preferences related to different classes of requirements. Secondly, requirements selection is based on predicting and rebalancing its impact on effort, time and quality. Both prediction and rebalancing uses the simulation model prototype GENSIM. Thirdly, alternative solution sets offered for decision-making are developed incrementally based on thresholds for the degree of importance of requirements and heuristics to find a best fit to constraints. Finally, trade-off analysis is used to determine non-dominated extensions of the maximum value that is achievable under resource and quality constraints. As a main result, quantitative Win-Win proposes a small number of possible sets of requirements from which the actual decision-maker can finally select the most appropriate solution.[20]

Factors Considered : Stakeholder preference ,effort, time and quality

16) Analytical Model for requirements selection Quality Evaluation[AMRSQE]

The model is based on queuing theory allowing calculations of attributes such as serving time, system load, and stability criteria. The selected requirements are evaluated and estimated with respect to expected market value and development effort. After screening, a fraction of the requirements are discarded and the remaining ones are propagated to evaluation. The requirements selected for evaluation includes both α and β requirements. The average evaluation effort is estimated to be the same for all requirements, regardless of quality. Finally, after disposing some of the evaluated Requirements, the construction phase is entered.[21]

Factors Considered : Market value, development
Effort, budget restrictions, requirement dependencies, requirement decomposition

17)Quality Performance Model (QUPER)

QUPER helps in setting the quality of a product in next release. It follows the following steps ,Define the quality indicators ,for each indicator and for each relevant estimate the breakpoints and barriers, estimate your product's current quality and the competing products quality ,estimate targets for coming releases, propose candidate targets, decide on actual targets, approve and communicate roadmaps as a common vision with realistic targets for downstream systems and software engineering and revise the roadmaps and iterate any necessary steps as estimates become more certain or circumstances changes.[22]

Factors Considered*:* Quality of non-functional requirements, Cost of non-functional requirements

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

18)A Mathematical Formalization for Flexible Release Planning (AMFFRP)

It is a mathematical formalization of release planning(AMFFRP) with a corresponding optimization tool that supports product and project managers during release planning. The tool is based on integer linear programming and assumes that an optimal set of requirements is the set with maximal projected revenue against available resources. The input for the optimization is twofold. The first type of input data concerns the list of candidate requirements, estimated revenues, and resources needed. Second, managerial steering mechanisms enable what-if analysis in the optimization environment.[23]

Factors Considered: the list of candidate requirements, estimated revenues, and resources needed

19)RP with Feature Trees

The model uses AND, OR, and REQUIRE dependencies to structure a solution's requirements as a feature tree. A feature tree simplifies the handling of a requirements specification in a release planning context. Features abstract from detail by grouping AND-related requirements. Allocating features instead of requirements to software releases reduces the number of release planning decisions. A feature tree hides incompleteness by handling non-specified features the same way as specified ones. A feature tree also captures requirements changes. feature trees] can be used to structure requirements and simplify release planning, hence to support release planning , i.e. the planning of variability over time[24]

Factors Considered: Stake holder preference, Product features, feature specification like effort and bugs, time

20) MAX-MIN Ant System with a Dynamic Roulette Wheel –Sofware Release planning.(MMASDRW-SRP)

The proposed algorithm, namely MMASDRW-SRP, adopts a dynamic roulette wheel strategy for giving a sophisticated balance between intensification and diversification, thereby improving the quality of solutions obtained.[25]

Factors Considered: Resource, stakeholder, urgency, and precedence and coupling relationships

21) Release Plan Simulator (REPSIM-1).

The model explains a three step method for release planning which can be added in addition to the existing steps. (i) strategic release planning that maps requirements to subsequent releases and (ii) a more fine-grained planning that defines resource allocations for each individual release (iii) stability analysis, which analyzes fine-grained plans of individual releases with regard to their sensitivity to planning errors. [26]

Factors Considered: Effort and recourse availability

22) Release Plan Simulator (REPSIM-1).

The model explains a three step method for release planning which can be added in addition to the existing steps. (i) strategic release planning that maps requirements to subsequent releases and (ii) a more fine-grained planning that defines resource allocations for each individual release (iii) stability analysis, which analyzes fine-grained plans of individual releases with regard to their sensitivity to planning errors. [27]

Factors Considered: Effort and recourse availability

23) RP with Fuzzy Effort Constraints(RPUFEC)

In this approach Fuzzy logic is used to handle the uncertainty of data regarding effort estimation, effort constrains and objectives related to cost, benefit and quality. Satisfaction of these constraints on effort is achieved by fuzzy system that focuses on satisfaction level of solution.[28]

Factors Considered: Requirement dependencies, Effort, Fuzzy constraints

24) Quality Improvement Paradigm (QIP)

It introduces a six step process model 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.[29]

Factors Considered: Requirement dependencies , effort, resource constraints

25)An Optimization technique for RP(AOTRP)

The model applies integer linear programming techniques to aid requirements managers of product software companies in release planning. The applied techniques take candidate requirements, estimated revenue per requirement (or combination of requirements), and available resources as input. The model is developed on the assumption that maximum revenue can be generated from a release by including the best set of requirements in a release. Planning suppleness is added by way of allowing flexibility in team composition, team transfers, extension of deadlines and hiring external resources [30]

Factors Considered*:* Requirement dependencies, projected revenue, and requirement resource claim per development team

26) Fuzzy Model for dependence constraints in RP(FMDCRP)

The model 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.[31]

Factors Considered: Structural constraints, Effort constraints.

27) Fuzzy Optimization Model for RP (FOMRP)

The model applies fuzzy theory to handle the uncertainty concerning dependency constraints from a holistic perspective, i.e. the whole set of fuzzy dependency constraints is considered as a fuzzy graph. The satisfaction of dependency constraints in a solution plan is measured by the distance between this plan and an ideal plan (in terms of the dependency constraints). The distance is materialized as the distance between two fuzzy graphs. This is considered to be an essential support for the actual decision-making.[32]

Factors Considered: Requirement dependencies, Effort, Resource

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

28) Consensus-Driven and Value based RP approach (CDVBRPA)

In first phase of this method requirements are prioritized according to value given by stakeholders. For identifying stakeholders‟ perceived value on requirements a following scale of one to five is used,(No-value ,Little value ,Some value High value ,Very high value).In second phase, a release configuration is selected by applying following steps. Identifying a configuration, Configuration assessment Decision on a configuration. Through above steps a release-configuration is identified, and then assessed by stakeholders to analyze estimated return on value. Finally a configuration is selected for implementation by consensus of all stakeholders [33]

Factors Considered: Time , Requirement dependency, Urgency

29) An Interactive and explanation oriented dialogue approach for RP

The approach contains the following steps: (1) Selection of a specific concern in a proposed solution; (2) Calculation of a stakeholder defined ideal plan; and (3) Comparing the actually generated plan and the prototype based on a similarity measure. The comparison of the actual and the ideal plan looks at aspects of interest for the stakeholder such as resource consumptions or structural properties of the plan. The proposed approach is generic and was applied and customized to three classes of wicked problems: release planning, investment planning, and urban planning.[34]

Factors Considered: Requirement precedence constraints, requirements coupling constraints, Pre assignment constraints, Effort estimation.

30) Post Release analysis of requirements Selection Quality (PARSEQ)

PARSEQ focuses on finding release planning process improvements through an analysis of earlier release planning decisions. The method aims at finding improvement suggestions for the release planning activity, as it is regarded as one of the most critical activities in market-driven software development .PARSEQ is divided into five steps: requirements sampling, re-estimation of cost and value, root cause analysis, elicitation of improvements, and prioritisation of improvements.[35]

Factors Considered: Precedence constraints, Resources, Stakeholder preference

31)Risk driven method for Extreme Programming (RDMXP_RP)

It is suitable for small teams, lightweight projects and vague/volatile requirements. It is a risk-driven method for XP release planning. Firstly, developers construct a set of feasible release plans from the project profiles which include those original ideas about the system's scope, cost, schedule, product quality and so on. Secondly, risks in each feasible release plan are analyzed. Risk analysis is used as the crucial tool when developers and customers plan releases. At last, stakeholders decide a certain release plan for the next iteration according with the result of risk analysis **[35]**

Factors Considered: Requirement dependencies, Value of requirement in terms of cost and revenue in terms of coast and revenue, cost of implementation, effort, Business value

32) Hybrid approach Incorporating CP with RP(RP&CP.)

The entire project, an RP project together with non-RP constraints, must be transformed into an equivalent CP expressed in the MiniZinc language. Non-RP constraints defined using the process are transformed naturally and the remaining, RP-only portion of the project to MiniZinc, is transformed using the developed tool. parameters which instruct the MiniZinc solver to use the assignments from the RP solution as a starting point in its search strategy are added to the code. The project is then solved using a CP solver.[36]

Factors Considered : Stakeholder scores, couplings between features, pre-assignment of features, linear resource constraints , soft and strict precedence constrains and non-RP constraints.

III. TAXONOMY OF SELECTION FACTORS USED IN SOFTWARE RELEASE PLANNING MODELS

A report of the results found through systematic review of RP models is presented below.

Release Planning Model	**Identified Factors**
CVA(Cost value Approach)	Stake holders preference , cost, value
IFM(Incremental Funding Method)	Cost, Effort, Time
EVOLVE (Evolutionary & Iterative Approach)	Requirement dependency, stakeholder priority, effort
EVOLVE+(Extension of Evolve)	Stakeholder preference, Effort estimate, precedence constraint, coupling requirements, resource constraints, risk
EVOLVE* (Evolve-Star)	inherent precedence, coupling and resource constraints , stakeholder preference, effort constraints, budget constraints
F-EVOLVE*(Finical- Evolve-Star)	Resource capacity constraint , Time constraints, Feature dependency constraints ,Cost, Revenue
EVOLVE^{ext} (Evolve Extended)	Requirement dependencies, Stakeholders value, Time to market , Requirements volatility
S-EVOLVE* (System- Evolve-Star)	Stakeholders‟ value, Stakeholders‟ satisfaction ,Technological constraints ,Resource consumption ,Capacity bounds on resources ,System's constraints

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

NRP(Next Release Problem)	Customers, Requirements (cost, requirement dependency) ,company budget
AHPSRP(Art &Science of Release Planning)	Stakeholder Preference, Resources(Time,Budget,Effort)
MORP (Multi Objective Release Planning)	Stakeholders preference ,cost, deadlines, resources, risk, requirement interdependencies
MONRP(Multi objective Next Release Problem)	Customer Preference, Resources
BORPES(Bi-Objective Release Planning for evolving systems)	Features selection based on SD-coupling, feature value based on business perspective stakeholder value
AEQWW(Quantitative WinWin)	Stakeholder preference ,effort, time and quality
AMRSQE(Analytical Model of Requirements Selection)	Market value, development Effort, budget restrictions, requirement dependencies, requirement decomposition
REPSIM-1(Release Plan Simulator)	Effort and recourse availability
QUPER(QUality PERformance) Model)	Quality of non-functional requirements ,Cost of non-functional requirements
AMFFRP (A Mathematical Formulation for flexible Release Planning)	Development by one pool of developers, Development teams , Team transfers , External resource or dead line extension ,Requirements dependency
RDMXP-RP (Risk-Driven Method for Extreme Programming)	Requirement dependencies, Value of each requirement in terms of cost or revenue, Cost of implementation, Effort per-iteration, Business value
PARSEQ(Post-Release Analysis of Requirements Selection Quality)	Precedence constraints, Available resources ,Stakeholder's needs ,Cost , Value
MMASDRW(MAX-MIN Ant System with a dynamic Roulette Wheel-Release Planning)	Resource, Stakeholder, Urgency, Precedence, Coupling relationships
RPUFEC(RP with fuzzy effort Constraints)	Effort , Resource
QIP (Quality Improvement Praradigm)	Requirement dependencies, Effort, Resource, Fuzzy Constraints
AOTRP (An Optimization technique for RP)	Requirement dependencies, Projected revenue, requirement resource claim per development team.
FMDCRP (Fuzzy Model for dependence constraints in RP)	Structural constraints and effort constraints
CDVBRPA(Consensus Driven value based RP approach)	Time, Requirement dependency, Urgency
FOMRP(Fuzzy Optimization Model)	Requirement dependencies, Effort, Resource
Dialogue approach in RP	Requirement precedence constraints, requirements coupling constraints, Pre assignment constraints, Effort estimation
RPFT (Release Planning with Feature Trees)	Stake holder preference, Product features, feature specification (effort, Bugs, Time)
RP&CP	Stakeholder scores, couplings between features, pre-assignment of features, linear resource constraints , soft and strict precedence constrains, non-RP constraints(Mutual exclusion, Additive synergy between features, Productivity investments)

IV. CONCLUSION

.An analysis of 32 most popular RP models was done. To this end, is presented the found strategic release planning models in an overview map and have created a taxonomy of requirements selection factors used . All these found models provide different solutions of strategic RP and discuss different requirements selection factors. It is observed that most important factors considered by these models in common are technical constraints, resource constraints, effort constraints (required effort) and stakeholder's influence in requirements selection.

REFERENCES

[1] .http://searchsystemschannel.techtarget.com/definition/release-management
[2] http://timelessmind.com/t/TME_releasemanager.html
[3] http://en.wikipedia.org/wiki/Maintenance_release
[4] http://www.spamlaws.com/windows-operating-system.html
[5] Hans Christian Benestand , Jo E Hannay "A Comparison of Model-based and Judgement based Release planning on Incremental Software Projects
[6] 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
[7] 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.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

[8] D. Greer and G. Ruhe. Software release planning: an evolutionary and iterative approach. Information and Software Technology, 46(4):243–253, 2004
[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-The, 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] 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
[28] : 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
[29] 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
[30] 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.
[31] 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
[32] 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
[33] 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.
[34] 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.
[35] 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.
[36] Mark Przepiora, Reza Karimpour, Guenther RuheA Hybrid Release Planning Method and its Empirical Justification, 2013.

Ms.Sandhia Valsala, holds a Master's degree in Computer Applications from Bharatiyar University, Coimbatore and is currently pursuing her PhD from Karpagam University Coimbatore.She has completed her Mphil Computer Science from Madhurai Kamaraj University.

Dr Anil R Nair holds a PhD in software reliability from Indian Institute of Technology Bombay, India. Currently he is Principal Scientist with ABB corporate Research, Bangalore, India. Before that he was manager-QRM with Deloitte consulting, Mumbai, India. He has published over 20 papers in international conferences and journals. He is concentrating on software engineering research with a specific focus on industrial applications Anil was Special session co-chair - Special Session on Modern Software Engineering Methods for Industrial Automation Systems, INDIN'2013, Special Session on Software engineering methods, tools and Practices for automation systems, ETFA 2013; Program Committee member - Third annual world conference on Soft computing-wcsc 2013, 3rd International Conference on Soft Computing for Problem Solving (SocProS-2013), India HCI, 2012. he lead organizer of 1st International Workshop on Modern Software Engineering Methods for Industrial Automation (MoSEMInA 2014). MoSEMInA 2014 is a co-located with International Conference on Software Engineering (ICSE 2014) held at Hyderabad, India..

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)