

An Efficient Goal Oriented Risk Analysis In Requirement Engineering

K.Venkatesh Sharma
JNTU Kakinada
venkatesh_k123@rediffmail.com

P V Kumar
Osmania University, Dept of CSE
pvkumar58@gmail.com

Abstract—In the field of software engineering, the requirement engineering is getting special attention as it is based on the stakeholder's interests. The main factors that a requirement engineering process considers are business requirements and user requirements. The requirements are used to enhance the development of the software product with low cost and the time it should satisfy all the requirements. We developed a modified Tropos goal model, in which the evidence of satisfaction and denial of the goal is calculated from the likelihood of the events corresponding to the goals. Relations are defined between multiple goals and events, which define the importance of a particular goal. The event may be considered as a risk according to their likelihood value. So the inter relation values of the goals and events gives the impact of that event on the particular goal. In order to analyze the risk in achieving some particular goals, a set of candidate solutions are generated. The candidate solutions are evaluated on the basis of a risk affinitive value of the goals according to their events. The risk affinitive value is calculated from the different set of risk parameters, which is set like high, medium and low. The risk parameters clearly evaluate the affinity of that event to a particular set of goals. This parameter reduces the limitation in the previous work, like occurrence of same cost to risk values. According to the proposed approach distinct cost to risk values are achieved. Finally, the candidate solutions with low cost are selected

Keywords—Requirement Engineering, Tropos Goal model, Candidate solutions, Goal layer, Event layer, Support layer

I. INTRODUCTION

Generally risk analysis is used for studying all the considerations, which lead to the failure of the program. It is a methods and techniques for documenting the impact of extenuation strategies [1] and for judging system criticality [2]. Risk analysis is also shown important in the software design phase to assess criticality of the system [3] where risks are examined and necessary steps are introduced. Usually, countermeasures correspond to a design, system fine tuning and then with a limited margin of change. However, it may happen that the risk reduction results in the revision of the entire design and possibly of the initial requirements, introducing thus extra costs for the project [4].

Requirements engineering is a process based method for defining, realizing, modeling, relating, documenting and maintaining software requirements in software life cycle that help to understand the problem better [5]. It has been shown that a large proportion of the publications in software development can be related back to requirements engineering

(RE) [6]. RE is the process of discovering the purpose in the software development, by identifying stakeholders and their needs, and documenting these in a form that is amenable to analysis, communication and subsequent implementation [7]. Failures during the RE procedure have a significant negative impact on the overall development process [8]. Reworking requirements failures may take 40% of the total project cost. If the requirements errors are found late in the development process, e.g. during maintenance, their correction can cost up to 200 times as much as correcting them during the early stages of the development process [9].

Adequate necessities are therefore essential to ensure that the system the customer expects is produced and that unnecessary exertions are avoided.

According to Goal-Oriented Requirements Engineering, analysis of stakeholder goals leads to substitute sets of functional requirements that can each accomplish these goals. These alternatives can be evaluated with respect to nonfunctional necessities posed by stakeholders. In the previous paper, they propose a goal-oriented approach for analyzing risks during the requirements analysis phase. Risks are analyzed along with stakeholder interests, and then countermeasures are identified and introduced as part of the system's requirements. This work extends the Tropos goal modeling formal framework suggesting new concepts, qualitative reasoning techniques, and methodological procedures. The approach is based on a conceptual framework composed of three primary layers: assets, events, and treatments.

In the field of software engineering, the requirement engineering is getting special attention as it is based on the stakeholder's interests. The main factors that a requirement engineering process considers are business requirements and user requirements. The requirements are used to enhance the development of the software product with low cost and the time it should satisfy all the requirements. One of the sensitive areas, which every software development process concentrate is the risk involved with the process. So, particular assessment measures have to be taken in order to minimize the risks in software development process. Yudistira Asnar and Paolo Giorgini [13] have proposed a method for risk analysis in requirement engineering. The method deals with a software development method called, Tropos Goal Model and with a Probabilistic Risk Analysis (PRA). Inspired from their work, we are planning to propose an approach on extending the

Tropos model with risk analysis feature. Tropos goal model consists of three layers, mainly Goal layer (GL), Event layer (EL) and Treatment layer (TL). The GL consists of set of goals that has to fulfill by the process and EL contains the constructs which helps to achieve the goals. The TL is working as the input, which helps in achieving the goals. We developed a modified Tropos goal model, in which the evidence of satisfaction and denial of the goal is calculated from the likelihood of the events corresponding to the goals. Relations are defined between multiple goals and events, which define the importance of a particular goal. The event may be considered as a risk according to their likelihood value. So the inter relation values of the goals and events gives the impact of that event on the particular goal. In order to analyze the risk in achieving some particular goals, a set of candidate solutions are generated. The candidate solutions are evaluated on the basis of a risk affinitive value of the goals according to their events. The risk affinitive value is calculated from the different set of risk parameters, which is set like high, medium and low. The risk parameters clearly evaluate the affinity of that event to a particular set of goals. This parameter reduces the limitation in the previous work, like occurrence of same cost to risk values. According to the proposed approach distinct cost to risk values are achieved. Finally, the candidate solutions with low cost are selected

II. RELATED WORK

Jac Ky Ang et al [10] has developed an expert system that has least focus on requirement engineering. In facts, requirement engineering is important to get all the requirements needed for an expert system. If the requirements do not meet the client's needs, the expert system is considered fail although it works perfectly. Currently, there are a lot of studies proposing and describing the development of expert systems. However, they are focusing in a specific and narrow domain of problems. Also, the major concern of most researchers is the design issues of the expert system. Therefore, we emphasize on the very first step of success expert system development – requirement engineering. Hence, we are focusing in the requirement engineering techniques in order to present the most practical way to facilitate requirement engineering processes. In this paper, they analyze expert system attributes, requirement engineering processes in expert system developments and the possible techniques that can be applied to expert system developments. Next, we propose the most appropriate techniques for the expert system developments based on the analysis. From this paper, a set of techniques for expert system development will be provided.

Lukas Pilat et al[11] have proposed an approach for problem in requirements engineering is the communication between stakeholders with different background. This communication problem is mostly attributed to the different “languages” spoken by these stakeholders based on their different background and domain knowledge. We experienced a related problem involved with transferring and sharing such knowledge, when stakeholders are reluctant to do this. So, they take a knowledge management perspective of

requirements engineering and carry over ideas for the sharing of knowledge about requirements and the domain. We cast requirements engineering as a knowledge management process and adopt the concept of the spiral of knowledge involving transformations from tacit to explicit knowledge, and vice versa. In the context of a real world problem, we found the concept of “knowledge holders” and their relations to categories of requirements and domain knowledge both useful and important. This project was close to become a failure until knowledge transfer has been intensified. The knowledge management perspective provided insights for explaining improved knowledge exchange.

Mina Attarha and Nasser Modiri [12] have adopted a critical and specific software systems last longer and they are ought to work for an organization for many years, maintenance and supporting costs of them will grow to high amounts in the upcoming years. In order to develop and produce special aimed software, we should piece, classify, combine, and prioritize different requirements, pre-requisites, co-requisites, functional and nonfunctional requirements (by using requirements engineering process, they can classify the requirements). Development and production of special software requires different requirements to be categorized (different requirements can be categorized using software requirements engineering). In other words, we have to see all requirements during the software's life cycle, whether they are important and necessary for our software at present time or they are not important currently but will become important in future. Requirements engineering aim is to recognize the stockholder' requirements and their verifications then gaining agreement on system requirements, is not just a phase completed at the beginning of system development not required any more, but includes parts of next phases of software engineering as well. To achieve this purpose, we acquired a comprehensive knowledge about requirements engineering. First, they defined requirements engineering and explained its aim in the software production life cycle. The main activities and purpose of each requirements engineering activity is described. Moreover, the techniques used in each activity are described for a better comprehension of the subject.

III. MOTIVATION BEHIND THE APPROACH

In the field of software engineering, the requirement engineering is getting special attention as it is based on the user's interests. The main factors that a requirement engineering process considers are business requirements and user requirements. The requirements are used to enhance the development of the software product with low cost and the time it should satisfy all the requirements. Similar to every software engineering process, the requirement engineering is also affected by the risk factors. The development of the software based on user requirements has higher risk factors. Recently, Yudistira Asnaret al. [13] have proposed a risk assessment in requirement engineering. The approach is characterized by a goal based software development methodology proposed by Tropos [13]. In their approach, a

specific Tropos methodology is used for the risk assessment. A goal-relation model is developed to find the most relevant goals to achieve and the risk and cost to achieve the same. Inspired from their research, we have plotted modified Tropos goal model for the risk analysis in the requirement engineering.

IV CASE STUDY

The case study includes the processing in a software development company for achieving maximum profit. The main processes in proposed case study are business development software (BDS), customer relation management software (CRM) and Web development applications (WDA). These service are the main feature of the proposed case study and these are supported by many other services like supply chain management (SCM), ERPs, etc. The case study is subjected to analyze the cost required to reach a target and the risk that can happen during the different level of the processing. The main aim of the three main features considered in the proposed case study is to earn money from the services. The case study is proposed by considering the scenario like a client want to develop software from the company. An in-depth analysis of different process happening in developing software detailed in the case study

Tropos is a software development methodology that adopts the concepts of agent goal, task, and resource and uses them throughout the development process [11], from early requirements analysis to implementation. Early requirements analysis model and analyze the organizational setting where the system-to-be will eventually operate. In the proposed approach, the Tropos goal- risk specified in [11] is adopted. I.e. the Tropos model is extended by adding constraints and relation in order to assess the risk factor. The Tropos GR model mainly consist three tuples, i.e. the number of node (N), number of relations (R) and uncertain events (U). Consider the following GR model.

The Tropos G-R model consists of mainly three layers, namely goal layer, event layer and support layer. The goal layer consists of goals, which are the needs that have to be achieved. The event layers consists of event nodes, which serves to achieve the goals and the bottom layer, the support layer, which contains the node which support either the event nodes or goal nodes.

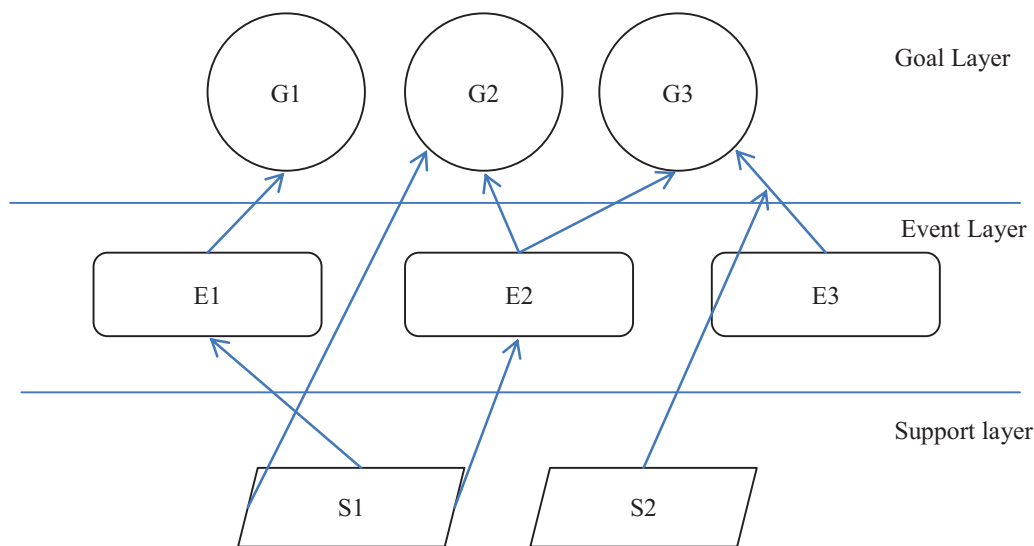


Fig 1. Tropos GR model

There are several uncertain events happen, which will restrict the successful achievements. Thus specific measures have to be taken to reduce the effect of these uncertainties. However, these measures imply extra costs for the whole process and they should be analyzed carefully before their adoption

A. Tropos Goal Risk model

Each of the three constraints is characterized by severity value and the severity is marked with four measure strong positive (++), positive (+), negative (-) and strong negative (--). The severity will be explained in the further section The constructs possess two attributes, satisfaction and denial, represented by SAT (c) and DEN (c), where c is the construct either goals, events and supports. The evidence of construct c will be satisfied for SAT(c) and denied DEN(c). In probability

theory, if $\text{Prob}(A) = 0.1$ then we can infer that probability of $\neg A$ is 0.9. Conversely, based on the idea of Dempster-Shafer theory [1], the evidence of a goal being denied (DEN) cannot be inferred from evidence on the satisfaction of the goal (SAT), and vice versa. For instance, the software development company has the goal to develop business development software, which is affected by the event `procurement_of_raw_materials`. The event may trigger the goal to either SAT() or to DEN() according to the support value. If the support user requirement has severity (--) then the goal result in Den (). The attribute values are specified more clearly by representing the value in different range like fully (*f*), partially (*p*) and none (*n*) and the priority of those values like $f > p > n$. The evidence for the satisfaction of a goal means that there is (at least) “sufficient” (“some”, “no”) evidence to support the claim that the goal will be fulfilled. Analogously, Full evidence for the denial of a goal means that there is “sufficient” evidence to support the claim that the goal will be denied. According to the severity the events and goals are listed and the SAT value and DEN value are calculated. The other feature that is concentrated on the proposed approach is the relationship between the goals, events and the support.

The relations *R* is the relations defined over different nodes in the defined goal risk model. The relation can be represented as $R = [N_1, \dots, N_n \mapsto N]$, where *N* is the target node and the N_1, \dots, N_n are the source nodes. The relations are defined as three types, decomposition relation, contribution relation and alleviation relation. The decompositions relation, which used are AND / OR, for refining the goals, events and supports. Contribution relation points the impact of one node to another.

Our Framework distinguishes four levels of contribution relations, ++, +, - and --. Each one of these types can propagate either evidence for SAT or DEN or both. For instance, the “++” contribution relation indicates that the relation propagates both SAT and DEN evidence, and the “++s” contribution relation means the relation only propagates SAT evidence toward target nodes.

Alleviation relations are similar to contribution relations but slightly differ in the semantics.

B. SDC Goal-Risk Model

The SDC means the software development company, which is considered in the case study proposed in the approach. The SDC goal model defines the three layers as goal layer, events layer and support layer. The goal layer contains the goals that have to achieve by the SDC

.As discussed above the events layer are helping parameters for achieving the goals. The support layer, as the name specifies provides support for the events as well as goals.

C. Goal Layer

The asset layer is adopted from the Tropos goal model [11] which analyzes strategic interests of stakeholders. In this layer of analysis, the goals of stakeholders are identified, refined,

and analyzed along with inter-relationships among them. In SDC, the main goals on concentration are earn money (G5), BDS- business development software (G1), CRM- Customer relation software (G6) and WDS- web development software (G9). Each of these nodes is AND/ OR decomposed into sub-nodes. For example, the node G5 is OR decomposed into nodes G1, G6 and G9. This decomposition will accelerate to fulfill the target goals.

The evidence of the goals provides the relevance of the sub nodes according to their SAT and DEN values. The SAT values indicate the source node has +ve impact on the target node, on the other hand DEN values indicates the -ve impact. After defining the decomposition relations, the contribution relations have to be defined. The contribution relation specifies how much a source node affects the root nodes. The contribution relations are specified by the +, ++, - and --. The + and - specified partial impact while ++ and - provides full impact. For example, the G2 and G3 give partial impact to G1, while G4 give full impact.

The figure 2 is the model of the SDC case study GR model. The following GR model depicts the total processing of the SDC under consideration. In the following figure, the target nodes can be listed as G5, G1, G6 and G9, while the source nodes are G2, G3, G4, G7, G8, G10, G11 and G12.

D. Event layer

The event layer is one of the deciding parameter in the SDC goal-risk model as an event can act both as positive feature and negative feature.

The negative feature is like the particular event can be act as a risk under certain conditions. Considering the figure 2, the event product cost (E4) has impact on the goal earn money (G5). If the product cost are less it act as profit value and the SAT (G5) will be full according to the E4, while the product cost is high then the impact will result in a risk thus results in the DEN (G5). The modeling of the event layer starts with event identification. There are different approaches for this, such as obstacle analysis [13], anti-goal [13], hazard analysis [13], misuse case [12], abuse case [13], taxonomy-base risk identification [11], or risk in finance [11]. Once events have been identified, they are decomposed into sub-events using similar decomposition relations as in the asset layer. This process continues until we reach leaf events that are easily observable.

TABLE 1. Likelihood calculation

| SAT(E) | DEN(E) | LH(E) |
|--------|--------|-------|
| F | N | L |
| F | P | O |
| P | N | O |
| F | F | R |
| P | F | R |
| P | P | R |
| N | F/P/N | U |

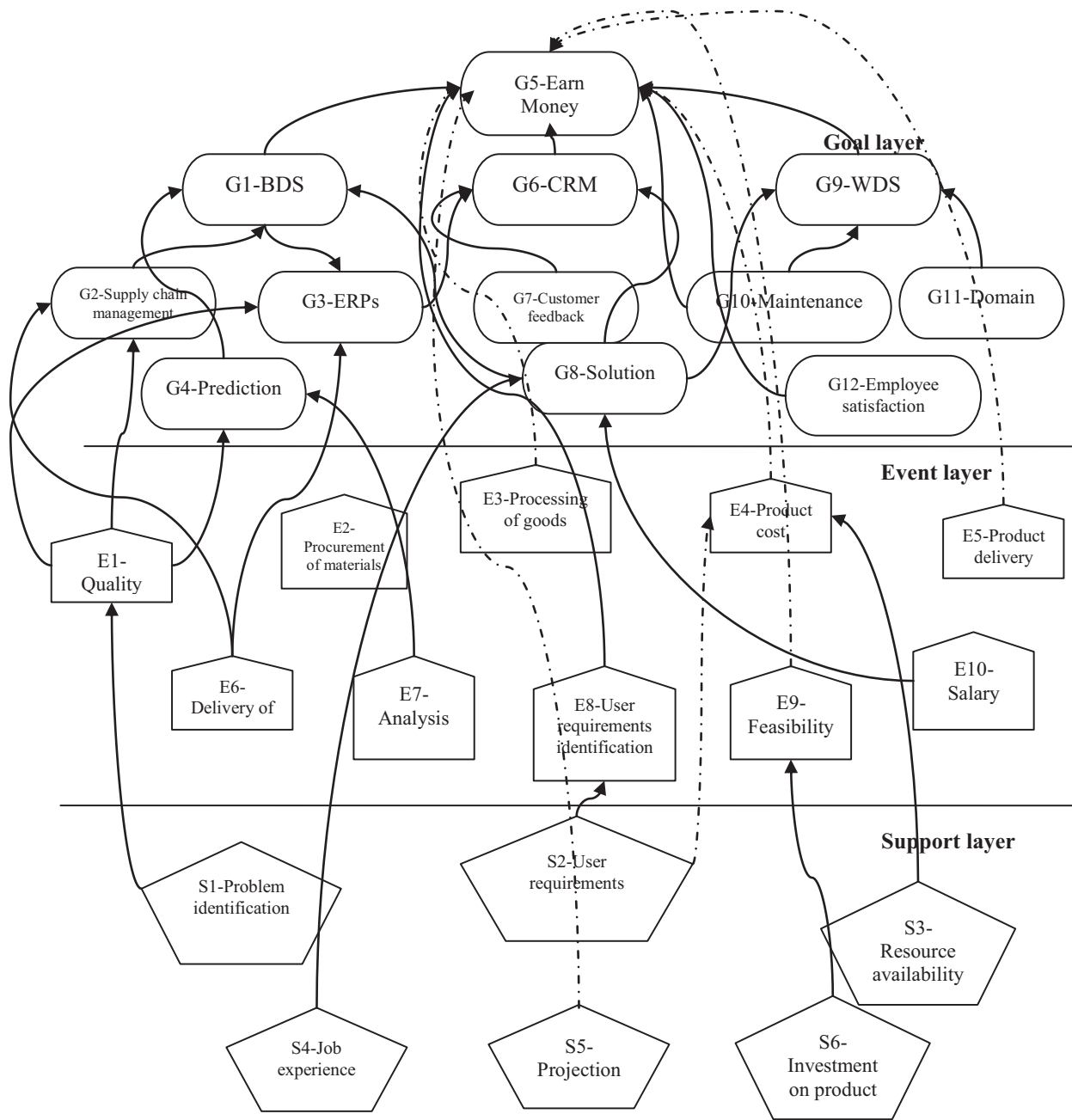


Fig 2. SDC goal- Risk Model

Once the events are identified, we have to characterize the events based on their likelihood and severity. The severity of the event can be assessed as the effect of an event to the achievement of a goal. As already mentioned, an event is characterized by two properties: likelihood and severity. In our framework, we calculate the likelihood of an event ($k(E)$) on the basis of the value of evidence that supports (i.e., SAT) and

prevents (i.e., DEN) the occurrence of the event. The likelihood is defined qualitatively and can take the following values: (L)ikely, (O)ccasional, (R)are, and (U)nlikely, with intended meaning $L > O > R > U$. Table 1 defines the calculation rules of likelihood from SAT and DEN values. The table prescribes an event with full evidence of being satisfied and no evidence of denial as a likely event. Consequently, an event without any evidence of satisfaction results to be an unlikely event, independently of any denial evidence. The severity is characterized by strong positive, positive, negative and strong negative. The positive and strong positive indicates that event is relevant or highly relevant. The negative and strong negative indicates that the event triggers to the goal denial. The response of the event characterizes, whether it is a risk or not. If an event triggers unexpected denial of the goal then it is considered as a risk. In our proposed approach, the likelihood and severity give specification to each event, with the help of that the event is judged for the risk assessment process. Consider the following scenario, In the below figure 3, two different cases of same node are depicted. One case is a satisfaction case and other is the denial case. In the first case, both source nodes are providing likely likelihood, so the evidence of the goal is SAT (G4) becomes high. On the other hand, in case 2 E7, the analysis conducted in an improper way, so the impact of the event definitely affects the goal G4.

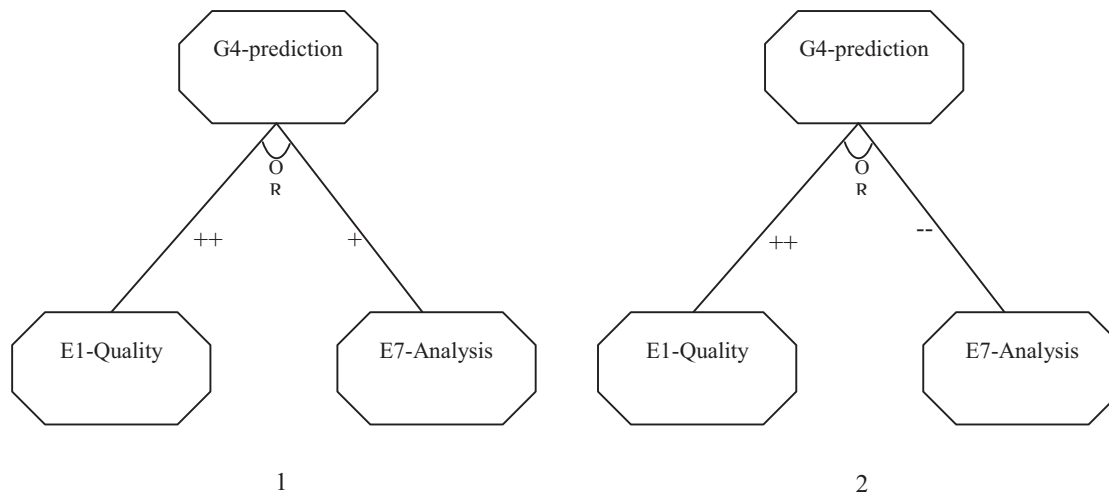


Fig 3. Event analysis

In case 2, the event E7 acts as risk and by identifying the likelihood and severity helps to reduce the effect of the risk caused by the event.

E. Support layer

The support layer, as its name specifies provides support to the event layer and the goal layer, the support layer also helps in mitigating the risks that affect in achieving the goal. The support value gives specification to the events and with the support parameter the likelihood and severity can be judged and reduced, if the event under consideration is acting like a risk. The support nodes are selected from the support layer based on specific events, which helps in reducing the risk caused by the event.

F. Risk Analysis methodology

In this section, we describe the methodological process and qualitative risk reasoning techniques used to analyze and evaluate alternatives in a GR model. Particularly, we focus on finding and evaluating all possible ways (called strategies) for satisfying top goals with an acceptable level of risk. In other words, given a GR model, each OR-decomposition introduces alternative modalities for top goal satisfaction, namely different sets of leaf goals that can satisfy top goals. Each of these alternative solutions may have a different cost and may introduce a different level of risk. Risk can be mitigated with appropriate countermeasures, which, however, may introduce additional costs and further complications. In the proposed risk analysis approach, two strategies are used for evaluating the different risk caused by different events.

The strategies are,

1. Cost analysis
2. Cost- Risk analysis

Based on the responses from the above listed analyses, the risks are mitigated to an accepting level with feasible cost for development. The analyses are detailed in the following sections. The analyses are based on the candidate solution

extracted from the different goals achievements. In achieving the goal G1, we can use anyone of the source nodes G2, G3 and G4. The case study discussed here has 4 target nodes; they are G5, G1, G6 and G9. The different candidate solutions for achieving the targets are represented with S. Consider following candidate solutions which are used for evaluating the target nodes.

S1- G2 G4 G7 G10 G11 G12
S2- G2 G3 G8 G10 G11 G12
S3 - G2 G4 G7 G8 G10 G11
S4- G2 G4 G7 G8 G11 G12
S5- G3 G4 G7 G10 G11 G12
S6- G3 G4 G8 G10 G11 G12
S7 – G2 G3 G4 G7 G10 G12
S8- G2 G3 G4 G8 G11 G12

Based on this candidate solution the risk analyses are subjected and the best solutions are selected and the rest are rejected.

G. Cost analysis

The cost analyses are conducted based on the generated candidate solutions, by considering their cost values. The candidate solutions are formed of six source node value. The cost analysis is subjected to extract the cost effective candidate solutions among the extracted candidate solutions. The cost is considered the impact of each process in the software development in the case of the SDC. The cost of a desired target is obtained by calculating the association of that node to the source nodes. Thus, the cost can be plotted as a set of three tuples,

$$Cost(Gn) = [Cost(G_{source}), Cost(E_n), Cost(S_n)]$$

Where, Cost (Gn) means the cost of the n the goal node, which is under consideration. Cost (G_{source}) is the number of source goal node which is supporting the target goal node. The values Cost (E_n) and Cost (S_n) are the cost for the event nodes and the support node. The cost value is also affected by the SAT values and DEN values of the nodes that are relevant for achieving the target goal. The other important factors are that affect the cost are the likelihood and severity value of the event node that give support the goals. Consider the following candidate solution,

S8- G2 G3 G4 G8 G11 G12,

Where the associations of different source goals are listed below,

G2→ SAT (P) E1 (++)→S1 (+)→E6 (-) = 10+5-5= 10;
G3→ SAT (F) E1 (++)→S1 (+)→E6 (+)= 10+5+5=20;
G4→ SAT (F) E1 (++)→S1 (+)→E7 (+)= 10+5+5=20;
G8→ DEN (F) E10 (--)→S4 (-) = 0
G11→ SAT (P) = 5

| | |
|--------------|------|
| G12→ DEN (P) | = 5 |
| COST (S8) | = 60 |

In the similar way, all other candidate solutions are set for obtaining their cost value in achieving the target goals. After calculating all the cost values, a cost graph is plotted and a threshold is fixed for the cost value. The threshold is set based on maximized SAT value and minimized DEN value. Thus, a number of candidate solutions are filtered out based on their SAT and DEN value. The filtered candidate solutions are then subjected to the next analysis phase of the risk analysis process.

H. Cost to Risk Analysis

The cost analysis proceeds to the cost to risk analysis. In this analysis phase the cost and risk of the candidate solution are evaluated. The candidates solutions considered in the cost to risk analysis is the filtered solutions from the cost analysis phase. This phase is initiated in order to analyze the risk affect for each of the candidate solutions. The nodes taking part in the each of the candidate solution are analyzed thoroughly.

The analysis considers the following parameters, chance_of_risk, chance_of_acceptance and chance_of_denial. The chance_of_risk is based on the evidence of likelihood and severity of event which triggers target goals. I.e. if the likelihood of the event is high, it will affect in achieving the goal. The event which provides the high likelihood is a risk then the target goal will result in denial. Similarly, chance_of_acceptance is related to the SAT () value and the chance_of_denial is based on the DEN () value. The total risk is calculated by assuming Null=1, Partial=2, and Full=3 and summing up the DEN values for all top goals. This means that for the acceptable risk level. A cost to risk graph is plotted for the assessment of relevant candidate solutions. The processing model can be depicted as following.

The figure 4, illustrates the working model of the cost to risk analysis of the proposed goal risk model. The above analysis separates the candidate solutions; those possess an acceptable risk measure and cost effectiveness.

V. Experimental Analysis

This section describes the experimental evaluation of the proposed goal risk model. The experiment is conducted in Java runtime environment in system configured to a processor of 2.1 GHz, 2 GB RAM and 500 GB hard disk. The experimental evaluations are provided in the following section. The proposed goal risk model is based on two and those analyses are used to judge the relevant candidate solutions

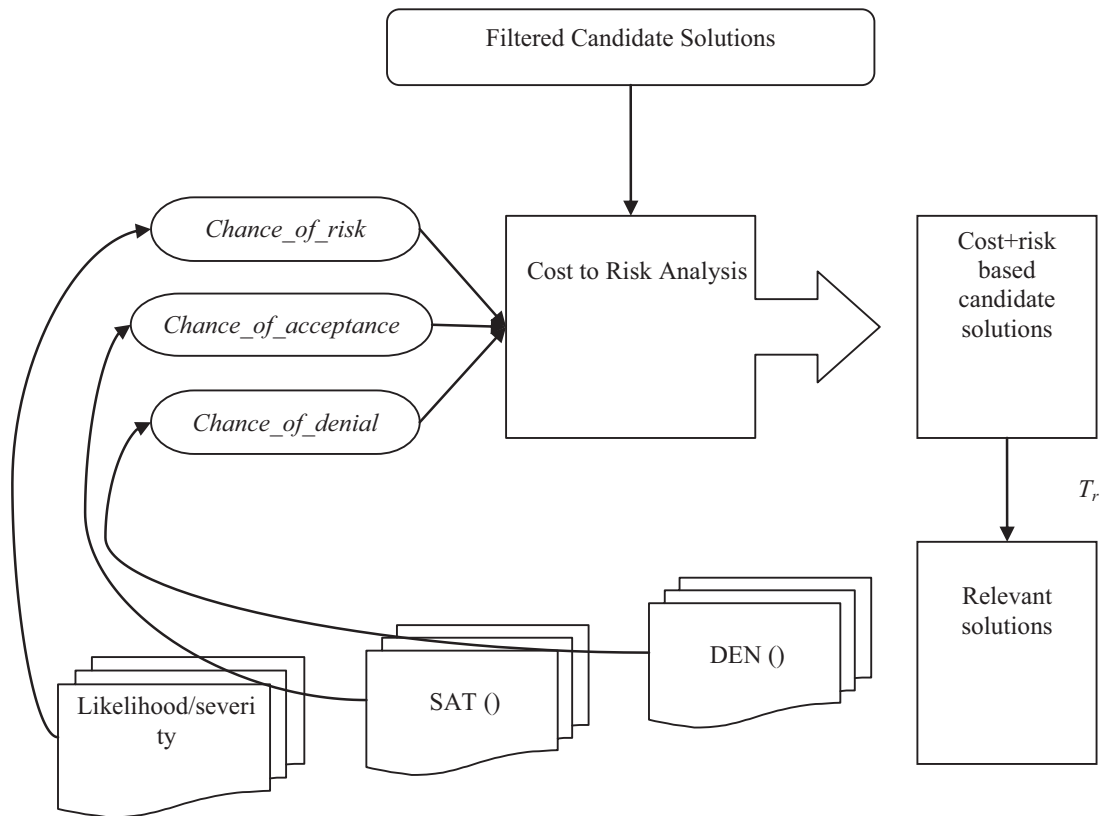


Fig 4. Cost to risk analysis

The experiments are carried out in the case study of the software development company, which has been detailed in the above section. The experimentation is evaluated based on the responses of the cost analysis and the cost to risk analysis. In the case of the SDC case study, there are mainly four target goals are present. The target goals are served by eight source nodes. The target goal nodes are G5- earn money, G1- BDS, G6- CRM and G9- WDS. The other nodes are listed as the sources that support the target goals the initial phase of the evaluation is to find the cost regarding the candidate solutions. The candidate solutions are calculated with help of a relevance analysis. The relevance calculation is based on the SAT, DEN and the likelihood values

The candidate solution generated are listed below,

- S1- G2 G4 G7 G10 G11 G12
- S2- G2 G3 G8 G10 G11 G12
- S3 - G2 G4 G7 G8 G10 G11
- S4- G2 G4 G7 G8 G11 G12
- S5- G3 G4 G7 G10 G11 G12
- S6- G3 G4 G8 G10 G11 G12
- S7 – G2 G3 G4 G7 G10 G12
- S8- G2 G3 G4 G8 G11 G12

The candidate's solutions shown above are selected for the cost analyses phase. The costs of each candidate solutions are calculated based on the association of source nodes, event nodes and the support nodes. The cost analyses plots a graph based on the cost values obtained. The cost of each node is calculated as per the SAT values ($F=10/P=5/N=0$) and DEN values ($F=0/P=5/N=10$). According to the above calculation the cost analyses is plotted as,

TABLE 2. Cost analysis

| Candidate solutions | Cost | SAT() | DEN() | Likelihood |
|---------------------|------|-------|-------|------------|
| S1 | 48 | F | N | L |
| S2 | 57 | P | N | O |
| S3 | 52 | F | P | O |
| S4 | 49 | F | P | O |
| S5 | 58 | P | F | R |
| S6 | 62 | N | P | U |
| S7 | 55 | P | P | R |
| S8 | 60 | P | P | R |

The table 2 represents the results of the cost analysis on the SDC. The analysis showed how much cost a solution requires for achieving the target goal. The fields of the table include the candidate solutions and the costs. The other three columns

refer to the supporting parameters for calculating the cost values.

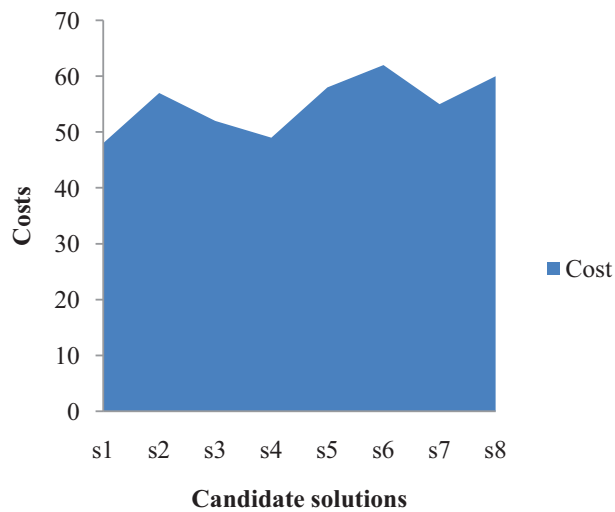


Fig 5. Cost analyses graph

The cost analysis plotted in the above graph, which ensures the total cost for achieving the target goals. A threshold is set based on the SAT and Likelihood values. The candidate solution which satisfies the threshold condition will be selected for the cost to risk analysis. In the current scenario, only S6 provides an “unlikely” likelihood, thus the threshold is set as those solutions, which is above than “unlikely” likelihood are selected for the cost to risk analysis. By doing this, a number of unwanted candidate solutions are restricted. This step also reduces the total cost and time required for execution. Thus the candidate solutions except S6 are selected for the cost to risk analysis.

In the cost to risk analysis phase, the cost and risk of the candidate solution for achieving the target goals are analyzed. The risk is calculated based on the DEN () value of the candidate solution under consideration. The denial rate of the candidate solution is based on the impact of events and support nodes of the solution. If the nodes are possessing high risk values or possessing high denial rate then the denial rate of the candidate solution will be higher. Consider the risk impact on the solution S3,

$$S3 \leftarrow G2 \ G4 \ G7 \ G8 \ G10 \ G11,$$

Where, G2, G7 and G11 having partial denial values. Thus the risk can be calculated as the sum of the evidence DEN (S3). The risk values are ranging from 3, 2 and 1 for full, partial and null denials respectively. Thus the risk of S3 can be given by, Risk (S3) $\rightarrow 2+2+2=6$, since DEN (G2) = DEN (G7) =DEN (11) = P. Similarly, the risks regarding all the candidate solutions are calculated and the graph is plotted based on the risk and cost values.

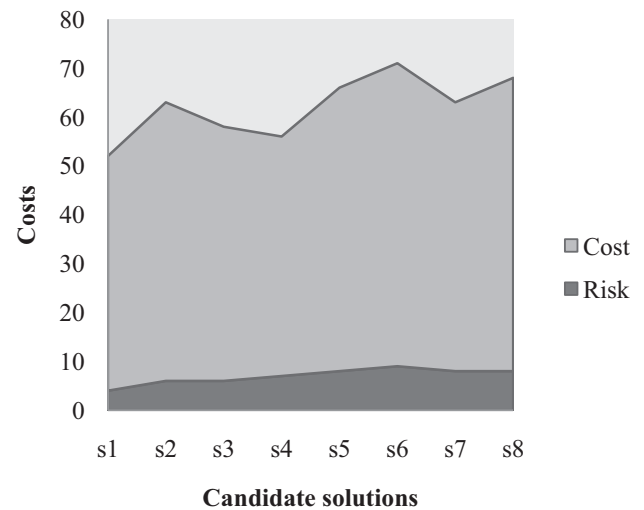


Fig 6. Cost to Risk analysis

The analysis of the graph plotted in figure 6, shows that the candidate solutions S1, S2, S3 and S4 are having acceptable costs and risk values. So for achieving the target goals, these solutions can be considered. Among the four solutions, the solution S3 is covering most of the source nodes with an acceptable cost with minimum risk. Thus we can state that the candidate solution S3 can achieve more profit with acceptable risk than other solutions.

VI. Conclusion

The requirement engineering is one of the emerging sections in software engineering. The requirement engineering is based on the user requirements and providing solutions according to it. We have proposed a requirement engineering model based on the Tropos goal model. A modified Tropos goal model is used in the proposed goal risk model. The goal risk model consists of three layers, and in the top level goals to be achieved by the process is plotted and in the second level, the events that triggers the goals and in the bottom level, the supporting parameters for the goal and events are plotted. The risk analysis of the proposed GR model is conducted based on two analyses, the cost analysis and the cost to risk analysis. The experimental evaluation is carried out on a case study considering a software development company. The results showed that the proposed goal risk model has attained solution with acceptable cost and risk. In future we are planning to introduce priority based parameter to the proposed approach to sort out the risk values. This will help to improve efficiency and also add more clarity to final results. The priority concept attempts more accuracy and acceptability in the final step. Thus we conclude of introducing goal oriented approach.

ACKNOWLEDGMENT

The Successful Completion of any task would be incomplete without expression of simple gratitude to the people who encouraged our work. Though words are not enough to express the sense of gratitude towards everyone who directly or indirectly helped in this task. We are thankful to this Organization TKR College of Engineering and Technology, Hyderabad which provided good facilities to accomplish my work and would like to sincerely thank our Chairman, Principal, Director, Guide and Faculty members for giving great support, valuable suggestions and guidance in every aspect of my work. We also thank Osmania University, Hyderabad for providing all help to perform all the required tests and come to conclusion. We also express our gratitude to all reviewers for giving us helpful comments and suggestions which has inspired us to keep the work moving forward for the benefit of society.

REFERENCES

- [1] Roy GG, Woodings TL (2000) A framework for risk analysis in software engineering. In: Proceedings of the seventh Asia-Pacific software engineering conference (APSEC '00), IEEE Computer Society Press, Washington, DC, USA, p 441
- [2] Boehm BW (1991) Software risk management: principles and practices. IEEE Software, pp.32–41. doi:10.1109/52.62930
- [3] B. W. Boehm. Software Risk Management: Principles and Practices. IEEE Software, pp.32–41, 1991.
- [4] Risk Analysis as part of the Requirements Engineering Process Yudistira Asnar Paolo Giorgini March 2007
- [5] Focusing on the Importance and the Role of Requirement Engineering Mina Attarha and Nasser Modiri Atta.mina@yahoo.com ;
- [6] H.F. Hofmann and F. Lehner, “Requirements engineering as a success factor in software projects”, *IEEE Software*, vol 18, no 4, pp. 58-66, 2001.
- [7] B.A. Nuseibeh and S.M. Easterbrook, “Requirements engineering: A roadmap”, Proc. of the 22nd Intl. Conf. on Software Engineering (ICSE '00), IEEE Computer Society Press, June 2000, pp. 35 – 46.
- [8] T. Hall, S. Beecham and A. Rainer, “Requirements problems in twelve software companies: An empirical analysis”, *IEEE Software*, vol 149, no. 5, pp. 153-160, 2002.
- [9] M. Niazi and S. Shastry, “Role of requirements engineering in software development process: An empirical study”, Proc. of the 7th Intl. Multi Topic Conf. (INMIC2003), IEEE Computer Society Press, Dec 2003, pp. 402-407.
- [10] Jac Ky Ang, Sook Bing Leong, Chin Fei Lee, Umi Kalsom Yusof, “Requirement Engineering Techniques in Developing Expert Systems”, School of Computer Sciences Universiti Sains Malaysia, IEEE, symposium on computer science and informatics, 2011, pp.1-2
- [11] Lukas Pilat and Hermann Kaindl, “A Knowledge Management Perspective of Requirements Engineering”, Institute of Computer Technology Vienna University of Technology Vienna, Austria, IEEE Conference: 19-21 May 2011, pp.1-12.
- [12] Mina Attarha and Nasser Modiri, “Focusing on the Importance and the Role of Requirement Engineering”, Interaction Sciences (ICIS), 2011 4th International Conference, 16-18 Aug. 2011, pp. 181 – 184.
- [13] Yudistira Asnar, Paolo Giorgini, “Risk Analysis as part of the Requirements Engineering Process”, Departmental Technical Report, 2007.