Cluster Computing (2019) 22:S14789–S14801
https://doi.org/10.1007/s10586-018-2417-3

CrossMark

Software engineering: risk features in requirement engineering

Shankar Nayak Bhukya[1] · Suresh Pabboju[2]

Received: 26 December 2017 / Revised: 1 March 2018 / Accepted: 6 March 2018 / Published online: 13 March 2018

Abstract

The term risk is defined as the potential future harm that may arise due to some present actions. Risk management in software engineering is related to the various future harms that could be possible on the software due to some minor or non-noticeable mistakes in software development project or process. There are quite different types of risk analysis that can be used. Basically, risk analysis identifies the high risk elements of a project in software engineering. Also, it provides ways of detailing the impact of risk mitigation strategies. Risk analysis has also been found to be most important in the software design phase to evaluate criticality of the system. The main purpose of risk analysis understands the risks in better ways and to verify and correct the attributes. A successful risk analysis includes important elements like problem definition, problem formulation, data collection. Some of the requirement risks are Poor definition of requirements, Inadequate of requirements, Lack of testing, poor definition of requirements etc. The likelihood of the events which tends to the goal can be evaluated from the evidence of Satisfaction and denial of the goal and it can be achieved through Tropos goal model. Original Tropos model is modified to meet the risk assessment requirements in requirements engineering. The event considers as a risk which based on the likelihood values. The relations are defined between multiple goals and events, which identify the necessity of a particular goal. In order to analyze the risk in achieving some particular goals, a set of candidate solutions are generated. Based on the risk affinitive value, the candidate solutions can be evaluated. There are three risk parameters to compute the risk affinitive value, which are (1) low (2) medium (3) high. The risk parameters and cost analysis clearly evaluate the affinity of that event to a particular set of goals.

Keywords Risk analysis · Risk management · Cost analysis · Tropos goal model

1 Introduction

Software engineering is the process is a systematic approach in developing software. It includes software development, operation and maintenance. Incompleteness and omissions of requirements caused many software hazards. As there is increased complexity in modern software systems it is essential to take necessary steps to avoid failure of systems [1]. Software systems might cause risk. Therefore risk assessment early in the life cycle of software development is very important. In fact risk assessment should be a part of risk management of software under development [2]. Asnar and Giorgini [3] believed that risk analysis should be part of requirement engineering process. Risk identification and risk mitigation procedures play an important role in software engineering.

The concept of goal is used as an emerging research area in software engineering to model early requirements and even non-functional requirements of a software system. The usage of goals help systems analyst to ascertain objectives of stakeholders and ensure that risk in requirements is identified and mitigated [3]. Tropos [4], GBRAM [5] and i* [6] are frameworks based on goal oriented approaches. The original Tropos is a software process model or methodology for developing agent-based software systems. In other words, Tropos is requirements-driven approach to software development. It exploits actor dependency analysis and goal analysis techniques for modelling software systems [7]. From the review of

✉ Shankar Nayak Bhukya
bsnaik546@gmail.com

Suresh Pabboju
plpsuresh@gmail.com

1 Faculty of Computer Science and Engineering, Osmania University, Hyderabad, Telangana, India

2 CBIT, Osmania University, Hyderabad, Telangana, India

literature it is found that Goal Risk model is useful in identifying risks in requirements engineering phase of a system. In addition to this, it is found that there is scope in the research area to investigate further utility in building good software and mitigate risk based on the goals of stakeholders.

In this paper, the Tropos goal risk model (TGR) and SDC goal risk model are presented before providing the proposed risk analysis using goal risk model. TGR model is the model containing layered approach in software risk analysis in the requirements engineering phase. Both are same with some differences in the risk analysis approach. However, the way they used layers remain same. The focus of every goal is to achieve something required by an organization or application. For instance, the goal "process loan" in banking sector is for processing load and generating revenue for bank. The main objective of the paper is to define a methodology to adapt Tropos methodology to have risk analysis in the requirements engineering phase of software development life cycle (SDLC). The advantage of the proposed system is to reduce risk and cost and help in choosing best solution. The following are the contributions of the paper.

1. We presented TGR model based on the original Tropos methodology. Our model contains various aspects of the model, relationships, truth tables, logical relationships and so on.
2. We proposed a methodology for risk analysis in the requirements engineering phase using the TGR model. The methodology facilitates production of various goal based solutions based on cost and risk. The analysis can help in finding best solution that reaches given goals incurring less cost and risk.
3. We implemented a prototype application that demonstrates proof of the concept. The application takes different goals, events and responses as input and produce different combinations of solutions to allow end user to pick the best one. The tool makes use of the proposed relationships and conditions to achieve this.

The remainder of the paper is structured as follows. Section 2 reviews literature related to risk analysis software requirements engineering. Section 3 presents TGR model. Section 4 presents SDC's (a software development company) goal risk model. Section 5 presents proposed risk analysis methodology. Section 6 provides experimental analysis. Evaluation of results is made in Sect. 7. Section 8 concludes the paper besides providing directions for future work.

2 Related works

This section reviews literature in the area of requirements engineering and risk analysis (RA) in that phase. Li and Duo [1] proposed a model to have systematic requirements safety analysis of software being developed. They considered V&V process model for empirical study and found that their model for software safety requirements analysis was effective. Sharma and Kumar [8] proposed a methodology based on Tropos Goal Model for optimization of solutions in the requirements phase. Appukkutty et al. [2] proposed a methodology for risk analysis of software at requirements engineering phase. They used Unified Modelling Language (UML) specification and mapped each requirement to an operational scenario found in UML. They focused on finding failure models and the possible causes. In Model Driven Engineering (MDE), Munante et al. [9] made a review on risk analysis with respect to security requirements engineering method. They used ISO 27005 criteria for risk analysis.

The goal oriented approach in software development and goal based requirements engineering framework [4] were extended by Asnar and Giorigini [3] by proposing a framework for goal-oriented approach for risk analysis in the requirements engineering phase. They considered Loan Origination Process (LOP) case study and applied TGR framework. Gupta and Sadiq [10] proposed Software Risk Assessment and Estimation Model (SRAEM) to predict risk of software projects in order to improve accuracy. Sharma and Kumar [11] proposed a Tropos based methodology for risk analysis in requirements engineering phase of software development. Islam and Houmb [12] presented a Goal Driven Software Development Risk Management Model (GSRM) which considers both non-technical and technical aspects into consideration with a holistic approach to integrate requirements engineering and risk management activities. Chriansen et al. [13] on the other hand propped a model known as Multiple Logistic Regression (MLR) to predict risk factor of software under development. Huzooree and Ramdoo [14] studied requirements engineering process across many organizations and found gap between ideal theoretical approaches and practices.

Sarigiannidis and Chatzoglou [15] proposed a conceptual framework meant for reducing risk in software projects. The factors they considered include project characteristics, risk identification approaches, risk dimensions, project risk management team, residual performance risk, project quality, and project performance. Martins and Gorschek [16] opined that in case of maintaining safety–critical systems, it is very important to focus more on requirements engineering. They made an empirical research with human participants using interview method along with secondary research. Sharma and Kumar [17] used modified TGR model for efficient risk analysis and find priority in requirements engineering phase. Based on the risk values prioritization parameter is included in the model. Asnar, Giorgini and Mylopoulos [18] proposed a risk assessment process based on TGR framework. The process takes acceptable risk, candidate solution criteria, events likelihood and severity, GR-model and input model as input and produce strategies that contain solutions and treatments. A software development company and its development scenario are considered by Sharma and Kumar [19] for an improved framework based on goal-risk model. From the literature it is understood that TGR model is suitable for risk analysis in requirements engineering. The existing work is found to be useful to have further investigations into the area. This is the motivation behind the work in this paper.

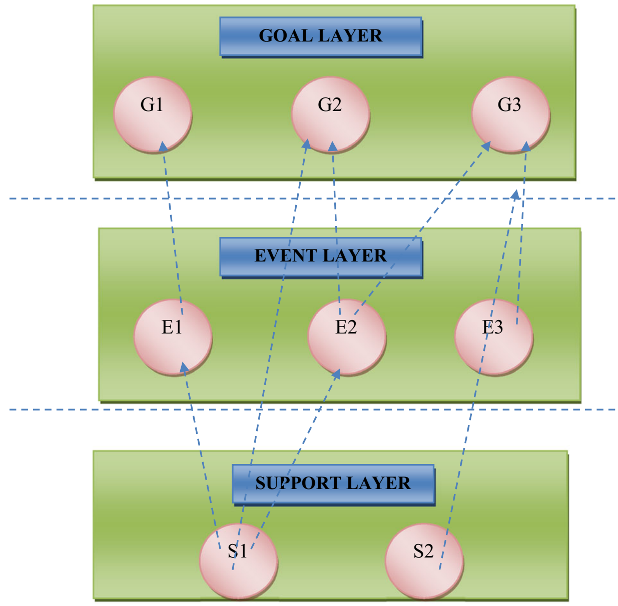

Fig. 1 TGR model

3 Tropos goal risk model (TGR)

Tropos is an agent-oriented software engineering methodology that covers the whole software development process. Tropos is based on two key ideas. First, the notion of agent and all related mentalistic notions (for instance goals and plans) can be used by all phases of software development, from early analysis down to the actual implementation. Second, Tropos covers also the very early phases of requirements analysis, thus allowing for a deeper understanding of the environment where the software must operate, and of the kind of interactions that should occur between software and human agents. The risk factor can be assessed by adding constraints and relations. The TGR contains three tuples, which are number of node (N), number of relations (R) and uncertain events (U).

The TGR model contains totally three layers, namely goal layer, event layer and support layer which are shown in Fig. 1. The goal layer consists of goal nodes, which consider as a target that needs to achieve. Subsequently, the event layer contains the event nodes which help to achieve the goals and the bottom layer. Similarly, the support layer contains the nodes; it supports both the goal layer as well as event layer. These three constrains are characterized by severity values and it is represented by four measures.

(i) Strong positive: $++$
(ii) Positive: $+$
(iii) Strong negative: $--$
(iv) Negative: $-$

Strong positive indicates that the occurrence of a specific event makes strong contribution towards goal satisfaction. Positive indicates that the occurrence of a specific event makes fair contribution towards goal satisfaction. Strong negative indicates that the occurrence of a specific event makes strong contribution towards goal denial. Negative indicates that the occurrence of a specific event makes fair contribution towards goal denial.

The constructs acquires two attributes, satisfaction and denial. Satisfaction is denoted as SAT while denial is denoted as DEN. In this context, satisfaction can be represented by SAT(c) and denial can be represented by DEN(c), where c is the construct of goals, events and supports. The evidence of construct c will be satisfied for SAT(s) and denied DEN(c). In probability theory, if Prob(A) = 0.1 then it infers the probability of A is 0.9. According to the Dempster–Shafer theory (DST), the evidence of a goal being denied (DEN) cannot be inferred from evidence on the satisfaction of the goal (SAT), and vice versa. “DST is a general framework for reasoning with uncertainty and understood connections to other frameworks such a probability, possibility, and imprecise probability theories.” For example, the ultimate aim of software Development Company is to develop business development software, which is affected by the event procurement_of_raw_materials. Depends on the support value, the event may trigger the goal to either SAT() or to DEN(). If the support user_requirement has severity (–), then the goal result is Den(). The attribute values can be represented by three different ranges such as fully (*f*), partially (*p*) and none (*n*), moreover, the priority of these values like $f > p > n$. In order to fulfill the goal, the evidence for the satisfaction of a goal means that there is (at least) ‘‘sufficient’’ (‘‘some’’, ‘‘no’’) evidence to support the claim. Similarly, the goals are not fulfilled which is denied, the evidence for the denial of the goal means that there is "sufficient" evidence to support the claim. The goals and events needs to plan or schedule as per the severity and then calculate the SAT value and the DEN value.

In the defined risk model, over the different nodes the relations are defined. The relation can be represented by

$R = [N_1, \ldots, N_n \mapsto N]$

where, R is the Relation, N is the target node, N_1, …, N_n are the source nodes.

As shown in Fig. 2, the relation can be classified into three types (i) decomposition relation (ii) contribution relation (iii) alleviation relation.

(i) Decomposition relation: The goals, events and supports can be refined by using AND/OR.
(ii) Contribution relation: Contribution relation points the impact of one node to another. The framework splits the contribution relation into four levels which are $+$, $++$, $-$ and $--$. Moreover, among these types, any one of it can propagate through SAT or DEN or both SAT and DEN. For example, "++" contribution specifies that the relation propagates through both SAT and DEN evidence. Likewise, " ++s" contribution indicates that the relation can only passes through the SAT evidence towards the target node.
(iii) Alleviation relation: The alleviation relations are similar to the contribution relation with a small difference in the semantics.

4 SDC goal risk model

SDC is the Software Development Company, this goal risk model defines the three layers which are goal layer, event layer and support layer. The SDC helps to achieve the goals in the goal layer. The event layers acts like a helping parameter to attain the goals; consequently, the support layer provides its support to both event layer as well as the goals.

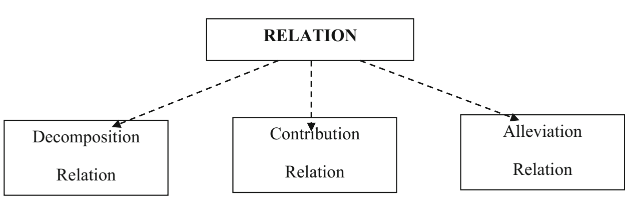

Fig. 2 Types of relation

4.1 Goal layer

The asset layer is adopted from the Tropos goal model which analyzes strategic interests of stakeholders, totally three processes is analyzed in this layer. Stakeholders are individuals or parties that are affected by the goal model. Strategic interests of stakeholders do mean that the natural and complementary interests of the stakeholders in the long run. Initially, the goal of the stakeholder is identified, refined and analyzed the inter-relationship among them. As shown in Fig. 3, the main goals focus on:

- G5-EM—earn money
- G1-BDS—business development software
- G6-CRM—customer relation software
- G9-WDS—web development software

According to Fig. 3, it is evident that G5 is the main goal which is the focus of the goal oriented approach in risk analysis in requirements engineering. Among all the nodes, each of nodes undergoes AND/OR process and decompose it into sub nodes. For instance, G5 is the main goal and it is decomposed into sub goals.

- Decompose the node G5 into G1, G6 and G9.
- This decomposition accelerates to accomplish the target goals.
- Depending on the SAT and DEN values, the evidence if the goal gives the significance of the sub-nodes.
- The values of SAT signifies the +ve impact for the source node on the target node.
- Similarly, the values of the DEN signifies the –ve impact for the source node on the target node,
- Define the contribution relations, after the decomposition relations are defined.
- The contribution relations denote how much a source node affects the root nodes.
- $+$, $++$, $-$ and $--$ indicates the contribution relation.
- $+$, $-$ indicates the partial impact.
- $++$, $-$ indicates the full impact.

The below GR model shows the entire process of SDC under consideration. In Fig. 3, the target nodes are G5, G1, G6 and G9 and the source nodes are G2, G3, G4, G7, G8, G10, G11 and G12.

4.2 Event layer

In SDC goal-risk model, the event layer can be used by the deciding parameter; moreover, this layer can be act as both positive feature and negative feature. The negative feature is like the particular event which is risk under certain conditions. The event 'E4' which is product cost in the

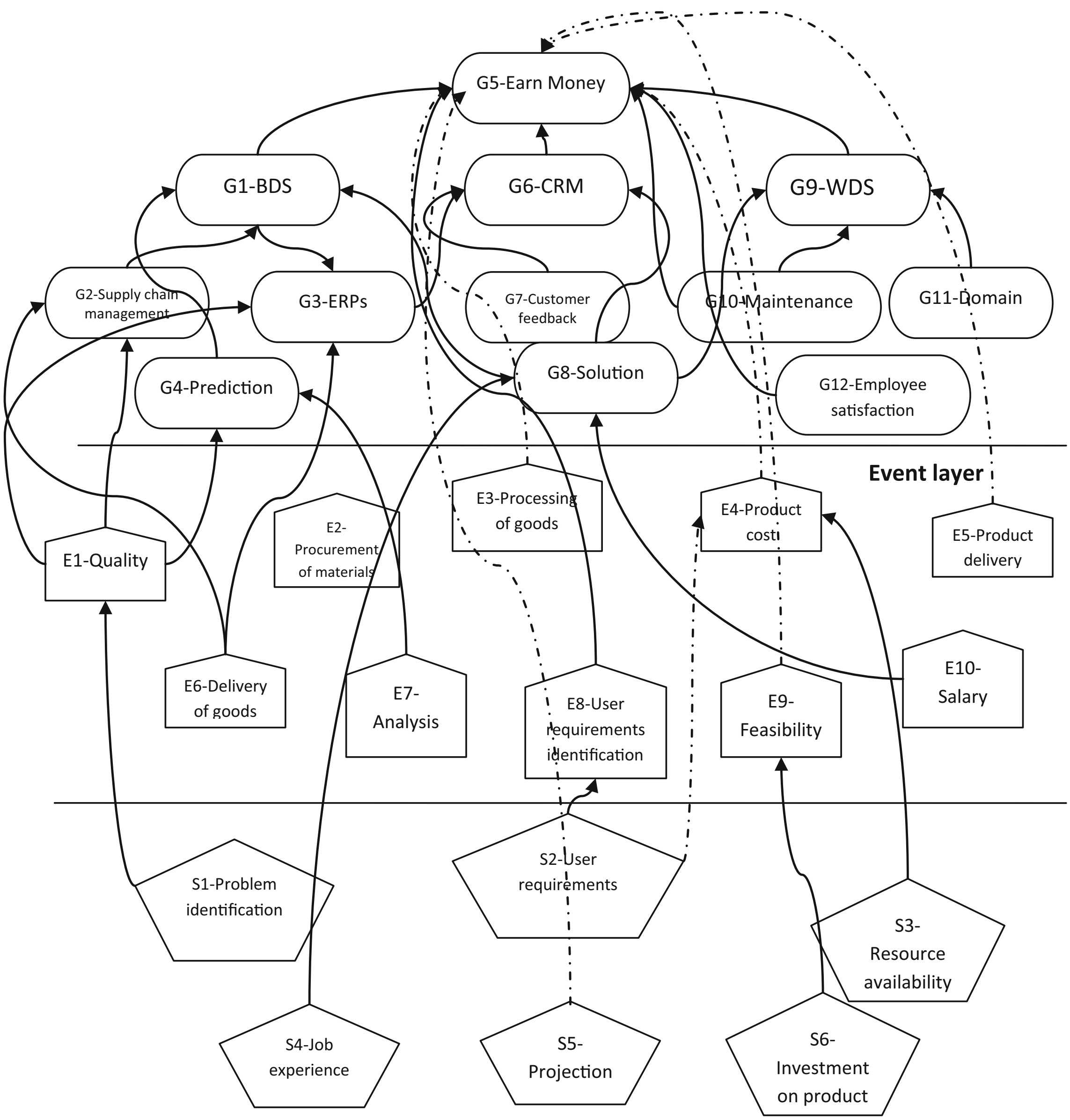

Fig. 3 SDC goal—risk model

event layer has the impact on the goal 'G5' which is earn money in the goal layer. Consider, as per Fig. 4

Product cost $\uparrow$ $-$profit $\downarrow$ $-$SAT(G5)

Product cost $\downarrow$ $-$profit $\downarrow$ $-$DEN(G5)

Product delivery $\uparrow$ $-$profit $\uparrow$ $-$SAT(G5)

Product delivery $\downarrow$ $-$profit $\downarrow$ $-$SAT(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). Similarly, if the product delivery is high, the profit is high which results of SAT (G5), if the product delivery is low, the profit is low which results of DEN (G5). Event identification is the initial process of modeling the event layer. Some of the different approaches are obstacle analysis, hazard analysis, misuse case, anti-goal, abuse case, taxonomy-base risk identification, or risk in finance. After the event identification, the sub-events can be obtained by decomposition process by using similar decomposition relations in the asset layer. This process continues until reaching the leaf events, which are easily observable.

An event can be characterized by two properties, which are likelihood and severity which is depicted in Fig. 5. After the event identification, the events are characterized on the basis of likelihood and severity. In order to achieve the goal, the effect of the event can be assessed by the severity of the event.

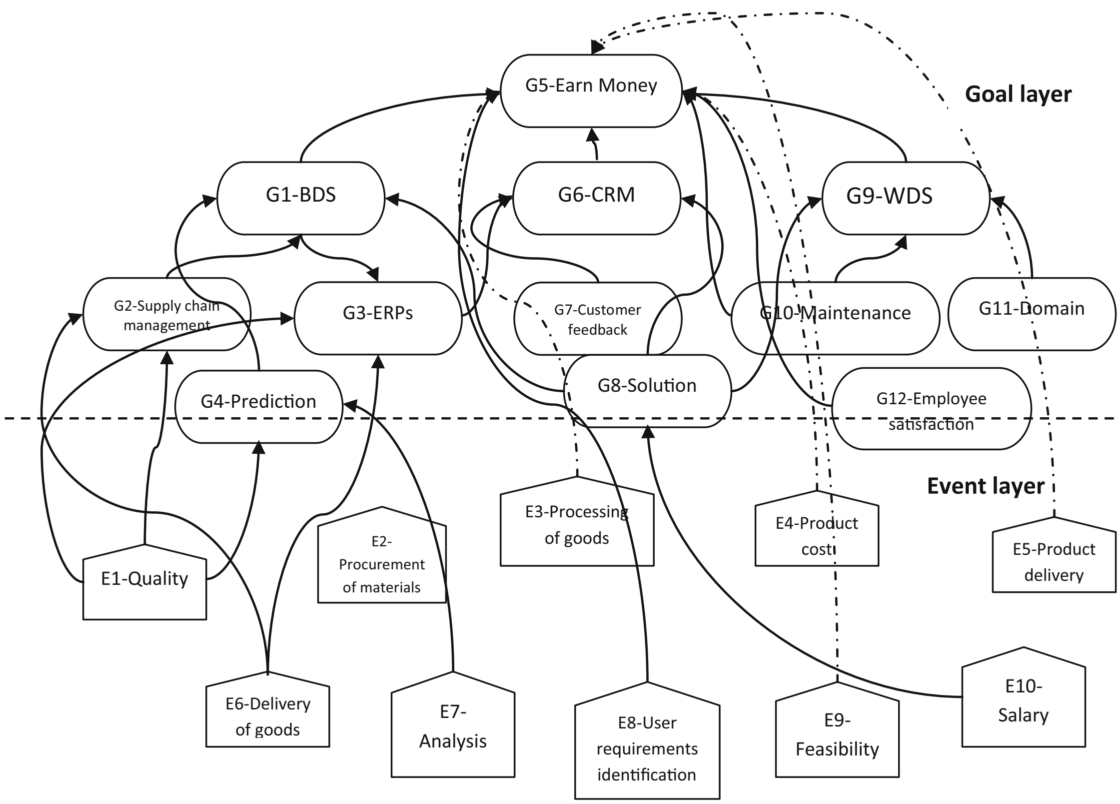

Fig. 4 SDC-goal and event layer

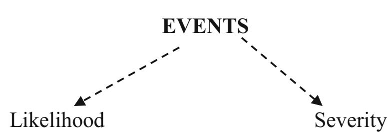

Fig. 5 Two properties of event

The likelihood of the event (k (E)) can be calculated on the basis of the evidence value that supports and prevents (SAT and DEN) the occurrence of the event. The likelihood can be defined by the following values Likely (L), Occasional (O), Rare (R) and Unlikely (U). AS per the priority, it can be represented by $L > O>R > U$. The likely function means, the event with full evidence for satisfaction (SAT) and no evidence for denial (DEN). Similarly, the unlikely function means, the event has no evidence for satisfaction (SAT). The four parameters which are used to characterize the severity are: positive, strong positive, negative and strong negative. Strong positive and positive indicates that the event is relevant or highly relevant and the strong negative a negative symbolizes the event triggers to the goal denial.

$++, + \rightarrow$ Event is relevant or highly relevant

$--, - \rightarrow$ Event triggers to denial of goal

The response of each event indicates that whether the event is risk or not. If an event triggers an unexpected denial of the goal then it acts as a risk. In the TGR model, for each and every event, the likelihood and severity provides specification, the event can be estimated by the risk assessment process through the likelihood and severity functions.

In Fig. 6a, b, two different scenarios are discussed. The Prediction (G4) is the goal; Quality (E1) and Analysis (E7) are the events. In case (a), the two source nodes provide likelihood; the evidence of the goal is SAT, so the G4 becomes high. In case (b), the analysis of the event E7 is not in a proper way (shown in (b) as '–' from G4 to E7), so the impact of the event affects the goal G4, due to this, the event E7 acts as a risk. Likelihood and severity can be used to reduce the effect of the risk caused by the event.

4.3 Support layer

The support layer provides support to the other two layers such as goal layer and the event layer. In order to achieve the goal, the support layer helps to mitigate the risks. The event gets its specification through the support layer; furthermore, with the help of support parameter, the likelihood and severity can be judged and reduced. The support

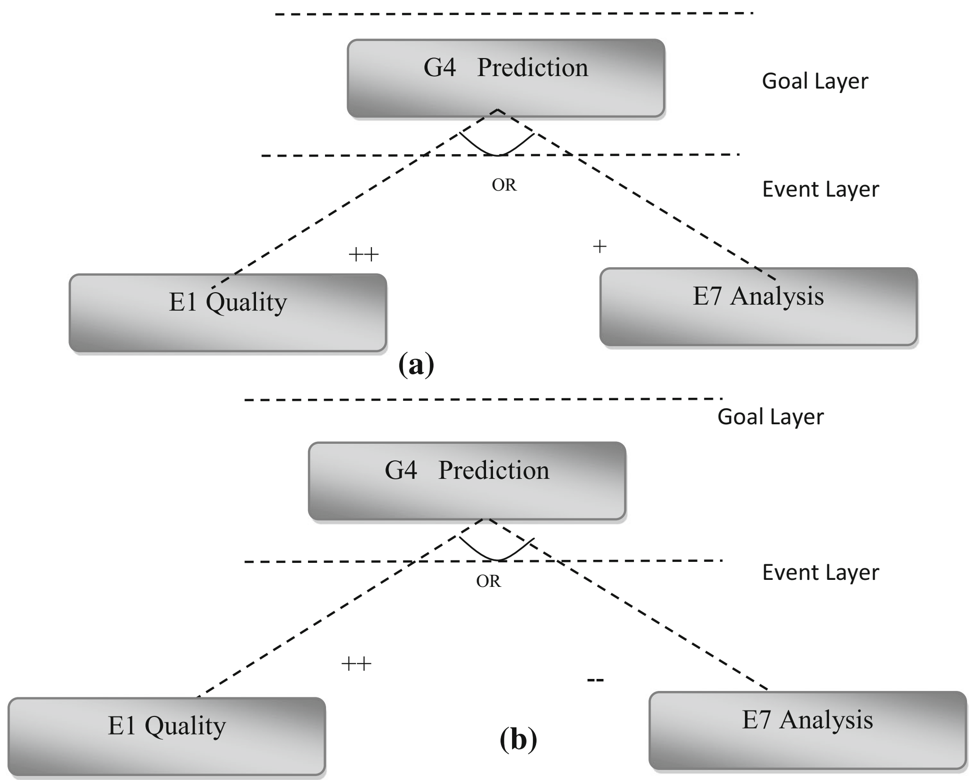

Fig. 6 Event analysis

layer contains the support nodes depends on particular event, which helps to reduce the risk present in the event.

5 Risk analysis methodology

The methodological process and qualitative risk reasoning techniques are preferable to analyze and evaluate the alternative in the goal risk (GR) model. All the possible ways also known as strategies can be used to find and evaluate to satisfy the top goals with some acceptable extends of risk. It has some other alternative methods in the GR model, each OR decomposition introduces alternative modalities for top goal satisfaction, namely different sets of leaf goals that can satisfy top goals. So, each of the alternative solution has its own cost and produces a different level of risks. The Risk can be mitigated by suitable countermeasures, which, however, may introduce additional costs and further complications. In this risk analysis approach, various risks from various events can be evaluated by two strategies. The strategies are:

- Cost analysis
- Risk prioritization
- Cost risk analysis

Depend on the responses from the above three analyzes, at feasible cost the risks are mitigate to acceptable level for development. These analyzes depends on the candidate solution which extracts from various goal achievements. According to Fig. 3, the target nodes are G5, G1, G6 and G9 and the target nodes are G2, G3 an dG4. The different candidate solutions are represented as ‘s’. For instance, the candidate solutions for evaluating the target nodes are given in Table 2,

- At first, subject the risk from the candidate solution.
- Select the best solution among them.
- Reject the remaining candidate solutions.

5.1 Cost analysis

The cost analyses can be conducted based on the generated candidate solutions while their cost values are also considered. From Table 1, each solution has six source node values. Furthermore, the cost analysis extracts the

Table 1 Likelihood calculations

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

candidate solution from the extracted candidate solution. In software development, the cost deems as the most impact of every process particularly in SDC. The target can be obtained by calculating the association of that node to the source nodes. So, the cost can be plotted as a set of three tuples,

$$Cost(Gn) = [Cost(G_{source}), Cost(E_n), Cost(S_n)] \tag{1}$$

where, cost (Gn) is the cost of ‘n’ goal node, Cost (G_{source}) is the number of source goal node, Cost (En) is the cost of event, Cost (Sn) is the cost of support.

Cost (Gn) is the cost of n goal nodes, which is under consideration. The Cost (G_{source}) is the number of source goal node which supports 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 SAT value and the DEN value of the nodes affect the cost values; these nodes are relevant for achieving the target goal. The other two important factors which affect the cost are likelihood and severity values of the event node and it provides support to the goals. Consider the candidate solution,

S8 - - - G2 G3 G4 G8 G11 G12

where the associations of different source goals are listed below,

support layer association. The goals have the association with both the cost as well as the risk. Here, a same element may provide both positive and the negative impact to achieve the goal, so, the risk identification is a quite tedious task. The goal with minimum cost and acceptable risk are preferable consideration while selecting the risk. The objects in each layer can be grouped to set of risks which depend on the SAT and DEN values, but, it is rare to calculate the risk by considering only these two parameters. So, the risk prioritization plays the role here. The risk prioritization can be calculated by probability method. The probability of an event to become a risk is calculated by the following equation,

$$probabilty(e) = \frac{n(DEN)}{n(DEN + SAT)} \tag{2}$$

Probability (*e*) is the probability of risk by an element ‘e’ in the tropos goal model, *n*(*DEN*) is the number of elements associated with element ‘e’ having DEN, *n* (*DEN* + *SAT*) is the number of elements associated with element ‘e’ having SAT and DEN.

According to the probability values of the elements, a list called risk priority list R_{list} is created, it is given by,

$$R_{list} = [e1, e2, \ldots, e3] \tag{3}$$

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

- By following the above process, the cost values for other candidate solutions are obtained to achieve the target goal.
- Plot the graph for the cost values.
- Fix a threshold value for the cost values.
- The threshold is set which based on maximum SAT value and minimum DEN value.
- Give the filtered solutions to the next analysis phase of the risk analysis process.

5.2 Risk prioritization

In TGR model, the cost analysis indicates the cost required for the goal achievement along with the event layer and the

The element assigns the risk level based on its priority and the SAT level of the elements associated with the element. The risk for particular event can be characterized by fixing a threshold which depends on the priority value and SAT level. An element is considered as risk by,

$$Risk(e) = \begin{cases} ++, & \text{if } probability(e) > threshold \text{ and } SAT(++) \\ --, & \text{if } probability(e) < threshold \text{ and } SAT(--) \end{cases} \tag{4}$$

As per the scenario, the risk prioritizing factor confirms the elements with most risk for the Tropos model.

- Differentiate the high risk and low risk elements according to the risk prioritizing process.

- The cost of the goal critically affects the higher risk elements.
- The elements with lower risk level are mitigated.
- So, the total number of elements can be reduced by risk prioritization process, which is consider as a risk and interns speed up the cost analysis phase.

Based on the risk values derived from the above expression, the priority list is given by,

$$R_{priority\_list} = [R(e1), R(e2), \ldots, R(e_n)] \tag{5}$$

$R_{priority\_list}$ is the cost to risk analysis phase, in order to reach the goal, the relevant risk can be found.

5.3 Cost to risk analysis

The cost analysis proceeds to the cost to risk analysis. The ultimate aim of this phase is to calculate the coat as well as risk. In this phase, the candidate solutions are from filtered solutions of the cost analysis phase. This analysis aims to analyze the risk affect for each of the candidate solutions. The candidate solution contains the nodes which are completely analyzed. The analysis considers the following parameters,

(i) Chance _ of _risk
(ii) Chance _ of _acceptance
(iii) Chance _ of _denial
(i) Chance _ of _risk: It is based on the evidence of likelihood and severity of event which triggers target goals. I.e. the goal can be affected when the likelihood of the event is high. The event provides the high likelihood is a risk then the target goal will result in denial.
(ii) Chance _ of _acceptance: It is related to the SAT value.
(iii) Chance _ of _denial: It is related to the DEN value.

The risk can be calculated by three parameters which are Null (N), Partial (P)and Full (F). So the total risk can be calculated by assuming Null = 1, Partial = 2 and Full = 3 and sum up all the DEN values to obtain the top goals, it is for all the acceptable levels. Furthermore, the candidate solutions can be assessed by plotting cost to risk graph. The cost to risk analysis process is showed in Fig. 6.

In Fig. 7, depicts the analysis which separates the candidate solutions and these possess an acceptable risk measure and cost effectiveness. The proposed methodology takes filtered candidate solutions as input. In addition to this SAT, DEN and likelihood details through chance of risk, acceptance and denial analysis are done for risk prioritization besides cost to risk analysis. The cost and risk analysis, as he names imply, are used to have better solution out of the candidate solutions. The resultant solution should be cost effective and with optimized or reduced risk. It is achieved using genetic algorithm we proposed earlier in our research paper [20].

6 Experimental analysis

The experiments are carried out in Fig. 3 which describes the scenario of Software Development Company for achieving maximum profit. The experimentation is computed based on the responses of the cost analysis and the cost to risk analysis.

6.1 Environment and results

This section explains the experimental evaluation of the goal risk (GR) 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.

As per the SDC consideration, it has totally four target goal nodes and these four target goal nodes are served by eight source nodes. The four target nodes are G5-earns money, G1-BDS, G6-CRM and G9-WDS. G2, G3, G4, G7, G8, G10, G11 and G12 are the source nodes which supports the above defined goal nodes. The initial phase of the evaluation is to find the cost for the corresponding candidate solutions. The candidate solution undergoes the process with the help of relevance analysis. The relevance analysis is based on the SAT, DEN and likelihood values. The candidate solutions generated are given 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 source node, event node and the support node helps to calculate the cost for the each candidate solution. The cost analyses plots a graph based on the cost values obtained. According to the SAT values, Full (F) = 10, partial (P) = 5 and None (N) = 0 and similarly the DEN values, Full (F) = 10, partial (P) = 5 and None (N) = 0 and depends these, cost values for the candidate solution can be calculated. The priority risk values can be mapped by three parameters which are High (H), Medium (M) and Low (L). Table 3 describes the cost analysis, as per the above calculation the cost analyses is plotted as.

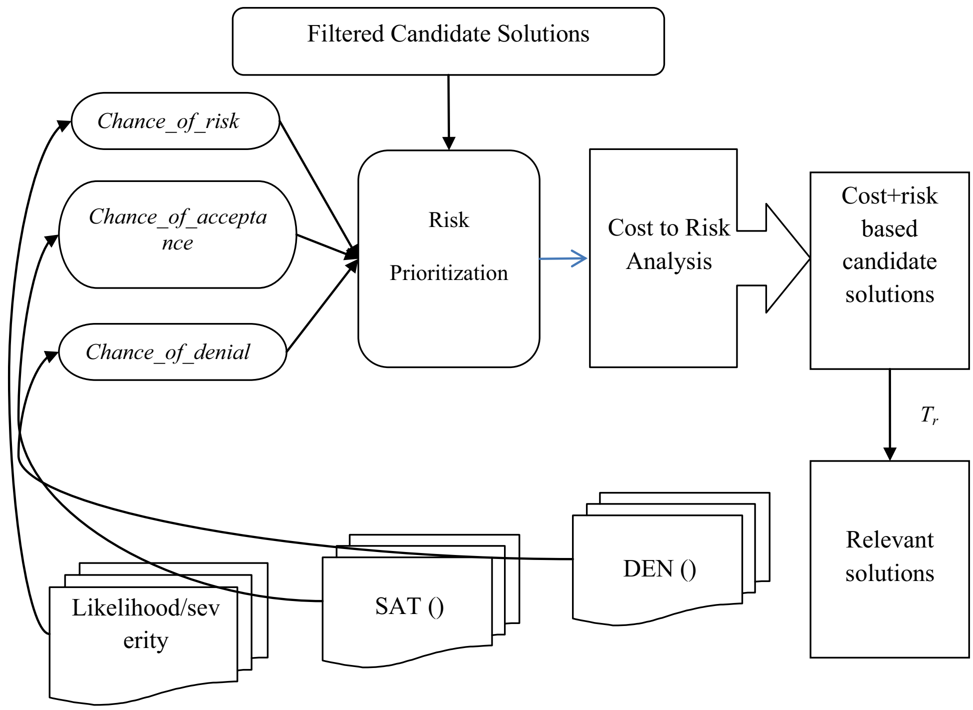

Fig. 7 Cost to risk analysis

Table 2 represents the results of the cost analysis on the SDC. The analysis shows that how much cost requires for the solution to achieve the target goal. The first two columns describe the candidate solutions and the costs. The other three columns show the supporting parameters for calculating the cost values. The risk priority values are numerical values, which are classified into three levels H, M and L according to the threshold level (Table 3).

The cost analysis plotted in Fig. 8, which ensures that the total cost for achieving the target goals. The threshold value which is based on the likelihood and the SAT values, moreover, the threshold value needs to satisfy the candidate solution, the satisfied candidate solution undergoes the cost to risk analysis process. Table 2 depicts that only S6 provides 'unlike' likelihood, so, the threshold is set as those solutions, which is above than "unlike" likelihood are selected for the cost to risk analysis. Due to this a number of unwanted solutions are restricted; it can be also useful in total cost reduction and the time requirement for execution purposes. The candidate solutions except S6 are eligible 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 provides high risk values or acquires high denial rate then the denial rate of the candidate solution will be higher. Consider the risk impact on the solution S3,

Table 2 Candidate solutions

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

S3 ← G2 G4 G7 G8 G10 G11

Here, the source nodes G2, G7 and G11 having partial denial values, thus the risk can be calculated by the sum of the evidence DEN (S3). The full, partial and null values can be assigned by 3, 2 and 1. So the risk of S3 can be given by, Risk (S3) → 2 + 2+2 = 6(because of the partial values), 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. In the cost to risk analysis, the risk priority value also with the risk calculation is incorporated. So, the incorporation of the risk priority value helps to reduce the level of risk by its priority. i.e., if a risk is calculated as 3 and another risk is calculated as 4, but if the risk with value 3 has a risk priority which maps as high and the risk with value 4 maps as low risk priority. Thus, as per the priority, the risk value 3 is taken as dominant risk while comparing with the risk value 4.

Table 3 Cost analysis

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

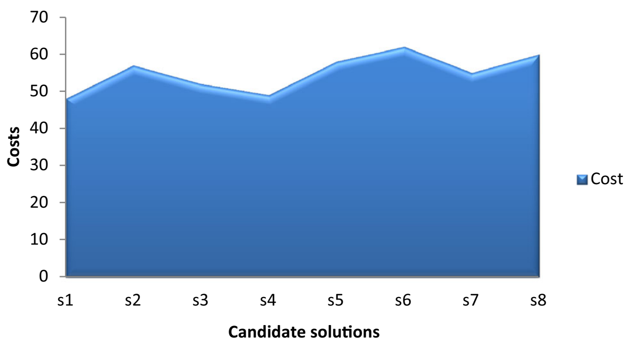

Fig. 8 Cost analysis graph

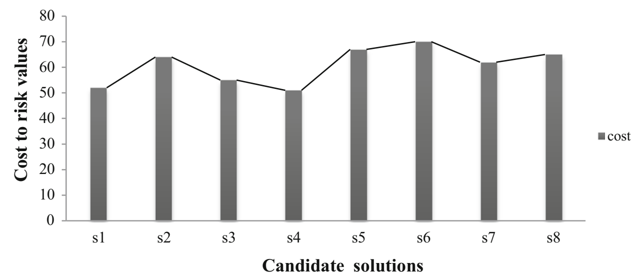

Fig. 9 Cost to risk analysis

The analysis of cost to risk analysis graph is plotted in Fig. 9, which shows that the candidate solutions S1, S2, S3 and S4 are having acceptable costs and risk values. In order to achieve the target goals, these solutions are taken into consideration. In these four solutions, the solution S3 is covering most of the source nodes with an acceptable cost with minimum risk. So, it declares that the candidate solution S3 can achieve more profit with acceptable risk than other solutions.

Figure 10 shows the risks and risk priority associated each risk. The risks can be either considered or discarded based on the priority values.

According to the SDC consideration, Z1 Z2 Z3 Z4 Z5 Z6 Z7 Z8 Z9 and Z10 solutions and X1 X2 X3 X4 X5 X6 X7 X8 X9 and X10 are the source nodes that provides support to the above defined goal nodes. The initial phase of the evaluation is to find the cost for the corresponding candidate solutions.

Z1 = X3 X4 X5 X6 X7 X8 X9 X10

Z2 = X3 X4 X5 X6 X7 X8 X9

Z3 = X4 X6 X7 X8 X9 X10

Z4 = X3 X6 X7 X8 X9 X10

Z5 = X1 X2 X3 X4 X5 X6 X7 X8

Z6 = X2 X5 X6 X7 X8

Z7 = X1 X2 X3 X4 X5 X6

Z8 = X2 X6 X7 X8 X9

Table 4 depicts the results of the cost analysis on the SDC. The analysis shows that how much cost requires for the solution to achieve the target goal. The first column describes the candidate solutions and the second column describes the costs for the corresponding candidate solution.

The cost analysis plotted in Fig. 11, which ensures that the total cost for achieving the target goals. As per the graph, at the candidate solution Z1 is high which means the cost is 122, and it is low at Z6 which is 71.

As can be seen in Fig. 11, it is evident that candidate solutions are presented in horizontal axis while vertical axis presents cost of the candidates. Based on the cost analysis, it is found that candidate solution named Z6 is the best solution when compared with other candidates.

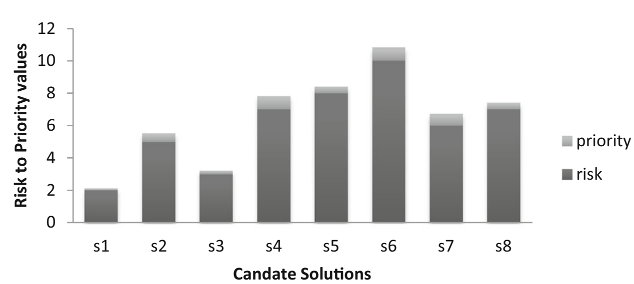

Fig. 10 Risk and risk priority comparison

Table 4 Cost analysis

Candidate solution	Cost
Z1	122
Z2	105
Z3	103
Z4	107
Z5	109
Z6	71
Z7	78
Z8	92

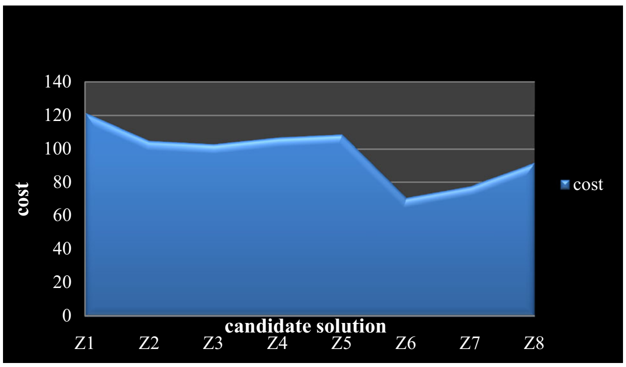

Fig. 11 Cost analysis graph

7 Evaluation of results

In this paper, the proposed model is evaluated with a case study. The analysis of results of the proposed model where our genetic algorithm [20] is employed revealed various facts. The observations are made in terms of cost, cost and risk and risk based priority of candidate solutions. Based on both cost and risk, the candidate solutions are analyzed to choose better solutions. Each solution is goal based solution that is best used to achieve main goal of stakeholders. Thus the proposed model has utility in producing best solutions. The results of the model are used in the requirements engineering phase in order to make well informed decisions. There are many existing works such as [3, 11, 17–19] that can be used to compare with our results.

8 Conclusion and future work

In software engineering, the requirement engineering is one of the emerging phases. There are quite different types of risk analysis that can be used. Basically, risk analysis identifies the high risk elements of a project in software engineering. Also, it provides ways of detailing the impact of risk mitigation strategies. Risk analysis has also been found to be most important in the software design phase to evaluate criticality of the system. The main purpose of risk analysis understands the risks in better ways and to verify and correct the attributes. A successful risk analysis includes important elements like problem definition, problem formulation, data collection. Some of the requirement risks are Poor definition of requirements, Inadequate of requirements, Lack of testing, poor definition of requirements etc. The likelihood of the events which tends to the goal can be evaluated from the evidence of Satisfaction and denial of the goal and it can be achieved through Tropos goal model. The event considers as a risk which based on the likelihood values. The relations are defined between multiple goals and events, which identify the necessity of a particular goal. In order to analyze the risk in achieving some particular goals, a set of candidate solutions are generated. Based on the risk affinitive value, the candidate solutions can be evaluated. There are three risk parameters to compute the risk affinitive value, which are (1) low (2) medium (3) high. The risk parameters clearly evaluate the affinity of that event to a particular set of goals. In future we intend to enhance our framework based on TGR model further and evaluate it with real world case studies besides using constraints in the goal model.

References

1. Li, S., Duo, S.: Safety analysis of software requirements: model and process. Rev. Procedia Eng. **80**, 153–164 (2014)
2. Appukkutty, K., Ammar, H., Goseva Popstajanova, K.: Software requirement risk assessment using UML. In: IEEE, pp. 1–4 (2005)
3. Asnar, Y., Giorgini, P.: Risk analysis as part of the requirements engineering process. Dit. Rev. **14**, 1–23 (2007)
4. Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J.: Tropos: an agent-oriented software development methodology. J. Auton. Agents Multi-Agent Syst. **8**(3), 203–236 (2004)
5. Anton, A.I.: Goal-based requirements analysis. In: Proceedings of the 2nd IEEE International Conference on Requirements Engineering (ICRE'96), p. 136. IEEE Computer Society Press, Washington, DC (1996)
6. Yu, E.: Modelling strategic relationships for process engineering. PhD thesis, University of Toronto, Department of Computer Science (1995)
7. Giorgini, P.:. Agent-Oriented Software Engineering Course, pp. 1–32. University of Trento, Trento (2010)
8. Venkatesh Sharma, K., Kumar, P.V.: A method to risk analysis in requirement engineering through optimized goal selection Tropos goal layer. J. Theor. Appl. Inf. Technol. Rev. **61**(2), 270–280 (2014)
9. Muñante, D., Chiprianov, V., Gallon, L., Aniorte, P.: A review of security requirements engineering methods with respect to risk analysis and model-driven engineering. In: International Conference on Availability, Reliability, and Security, pp. 1–16 (2016)
10. Gupta, D., Sadiq, M.: Software risk assessment and estimation model. In: International Conference on Computer Science and Information Technology, pp. 963–967 (2008)

11. Venkatesh Sharma, K., Kumar, P.V.: A method to risk analysis in requirement engineering using Tropos goal model with optimized candidate solutions. Int. J. Comput. Sci. Rev. **10**(6), 250–259 (2013)
12. Islam, S., Hilde Houmb, S.: Integrating risk management activities into requirements engineering. In: IEEE, pp. 1–12 (2010)
13. Christiansen, T., Wuttidittachotti, P., Prakancharoen, S., Arjong Vallipakorn, S.: Prediction of risk factors of software development project by using multiple logistic regression. ARPN J. Eng. Appl. Sci. **10**(3), 1324–1331 (2015)
14. Huzooree, G., Devi Ramdoo, V.: A systematic study on requirement engineering processes and practices in mauritius. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **5**(2), 40–46 (2015)
15. Sarigiannidis, L., Chatzoglou, P.D.: Software development project risk management: a new conceptual framework. J. Softw. Eng. Appl. **4**, 293–305 (2011)
16. Eduardo, L., Martins, G., Gorschek, T.: Requirements engineering for safety-critical systems. In: IEEE Software, pp. 49–57 (2017)
17. Sharma, V.K., Kumar, P.V.: An efficient risk analysis based risk priority in requirement engineering using modified goal risk model. Int. J. Comput. Appl. **73**(14), 15–25 (2013)
18. Asnar, Y., Giorgini, P., Mylopoulos, J.: Goal-driven risk assessment in requirements engineering. Requir. Eng. **16**, 101–116 (2011)
19. Sharma, V.K., Kumar, V.P.: An efficient risk analysis in requirement engineering. In: IEEE, pp. 1–5 (2012)
20. Bhukya, S.N., Pabboju, S.: An effective risk analysis in requirements engineering using modified goal model. Int. J. Pure Appl. Math. **118**(18), 1401–1418 (2018)

Shankar Nayak Bhukya has been working as a Associate Professor and HOD, Department of Information Technology, Swami Vivekananda Institute of Technology, Hyderabad, Telangana State, India. He has 12 years of teaching experience. Graduated from JNTUH and Post Graduated from IIT MADRAS, Chennai. He is Pursuing Ph.D. degree in CSE, research work in the area of software engineering from Osmania University.

Suresh Pabboju has been working as Professor and HOD, Department of Information Technology, Chaitanya Bharathi Institute of Technology. He has 31 years of teaching experience. Graduated from CBIT, Osmania University, and Post Graduated from JNT University, Hyderabad. He was awarded Ph.D. from Osmania University, Hyderabad, Telangana, India.