# A Method to Risk Analysis in Requirement Engineering Using Tropos Goal Model with Optimized Candidate Solutions

**K.Venkatesh Sharma[1], Dr P.V.Kumar[2]**

[1] **Research Scholar in JNTUK**
**Kakinada, Andhra Pradesh, India**

[2] **Professor in CSE Department, Osmania University,**
**Hyderabad, Andhra Pradesh, India**

**Abstract**

The requirement engineering is a field, in which software are modeled according to the requirements of the user. The software developed under requirement engineering will satisfy the users mostly on their perspective. So, recent researches are concentrating on the software development and analysis based on requirement engineering. The requirement engineering processes are also challenged by the risks in developing the software. So an efficient risk analysis system and risk management system is inevitable for the software development process under requirement engineering. In the proposed approach, a modified Tropos goal model for tackling the risk associated with the software development cycle is adopted. The Tropos goal model is three layer goal models, the top layer is the goal layer, the event layer in the middle and support layer in the bottom. The proposed approach defines a method to analyze the association between the nodes of each layer to evaluate their chances of raising the risks. The evaluations are assessed based on the nodes in the goal layer. On thorough analysis on the associations and impacts of event layer and support layer nodes to the goal layer nodes, the chances of raising risk can be calculated. The proposed approach explores the relative parameter like satisfaction parameter and denial parameter to efficient analysis of the risk factor. The experiments are conducted on java programming under JDK 1.7.0 platform and detailed analysis section is provided to find the cost to risk measure.

***Keywords:*** *Requirement Engineering, Tropos Goal model, Candidate solutions, Goal layer, Event layer.*

## 1. Introduction

Generally risk analysis is used for studying all the considerations, which lead to the frailer 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 non functional 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. YudistiraAsnar 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.
The main contributions of the paper are,

- A goal oriented approach is furnished to analyze the cost and risk associated with requirement engineering
- A genetic algorithm is utilized to optimize the candidate solutions

The rest of the paper is organized as; section 2 describes the literature survey regarding the requirement engineering and risk analysis. The 3rd section contains the problem description behind in proposing the approach. The 4th section includes the proposed goal model and case study used for it to analyze the risks and costs in requirement engineering. The 5th section consists of the experimental analysis of the proposed goal model. Finally, the 6th section includes the conclusion of the work.

.

## 2. Literature Review

The following section describes review about some recent works regarding the requirement engineering and risk analysis related to it. Security risk assessment in the requirements phase is challenging because probability and damage of attacks are not always numerically measurable or available in the early phases of development. Selecting proper security solutions is also problematic because mitigating impacts and side-effects of solutions are not often quantifiable either. In the early development phases, analysts need to assess risks in the absence of numerical measures or deal with a mixture of quantitative and qualitative data. GolnazElahietal[14] propose a risk analysis process which intertwines security requirements engineering with a vulnerability-centric and qualitative risk analysis method. The method is qualitative and vulnerability-centric, in the sense that by identifying and analyzing common vulnerabilities the probability and damage of risks are evaluated qualitatively. They also provided an algorithmic decision analysis method that considers risk factors and alternative security solutions, and helps analysts select the most cost-effective solution. The decision analysis method enables making a decision when some of the available data is qualitative.
JacKyAng*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 clients' 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. They have analyzed expert system attributes, requirement engineering processes in expert system developments and the possible techniques that can be applied to expert system developments.
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 non functional 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.

## 3. Problem Description

The requirement engineering is a field, in which software are modeled according to the requirements of the user. The software developed under requirement engineering will satisfy the users mostly on their perspective. So, recent researches are concentrating on the software development and analysis based on requirement engineering. The requirement engineering processes are also challenged by the risks in developing the software. So an efficient risk analysis system and risk management system is inevitable for the software development process under requirement engineering. In the approach, a modified Tropos goal model for tackling the risk associated with the software development cycle is proposed. The Tropos goal model is three layer goal models, the top layer is the goal layer, the event layer in the middle and support layer in the bottom. The proposed approach defines a method to analyze the association between the nodes of each layer to evaluate their chances of raising the risks. The evaluations are assessed based on the nodes in the goal layer. On thorough analysis on the associations and impacts of event layer and support layer nodes to the goal layer nodes, the chances of raising risk can be calculated. The proposed approach explores the relative parameter like satisfaction parameter and denial parameter to efficient analysis of the risk factor. The important factor about the proposed approach is that, it gives preference to the relationships between the different layers of the Tropos goal model

### 3.1 Tropos Goal model

Tropos goal model is a software development model, which is characterized by concepts of agent goal, task, and resource and uses them throughout the development process from early requirements analysis to implementation. Early requirements analysis model provides the organizational settings, where the system-to-be will eventually operate. 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). Considering a Goal Risk (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. 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 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(s) and denied DEN(c).In probability theory, if Prob(A) = 0.1 then we can infer that probability of ¬A is 0.9. Conversely, based on the idea of Dumpster-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 called *business development software*, which is effected by the event that needs to be given importance called *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. 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

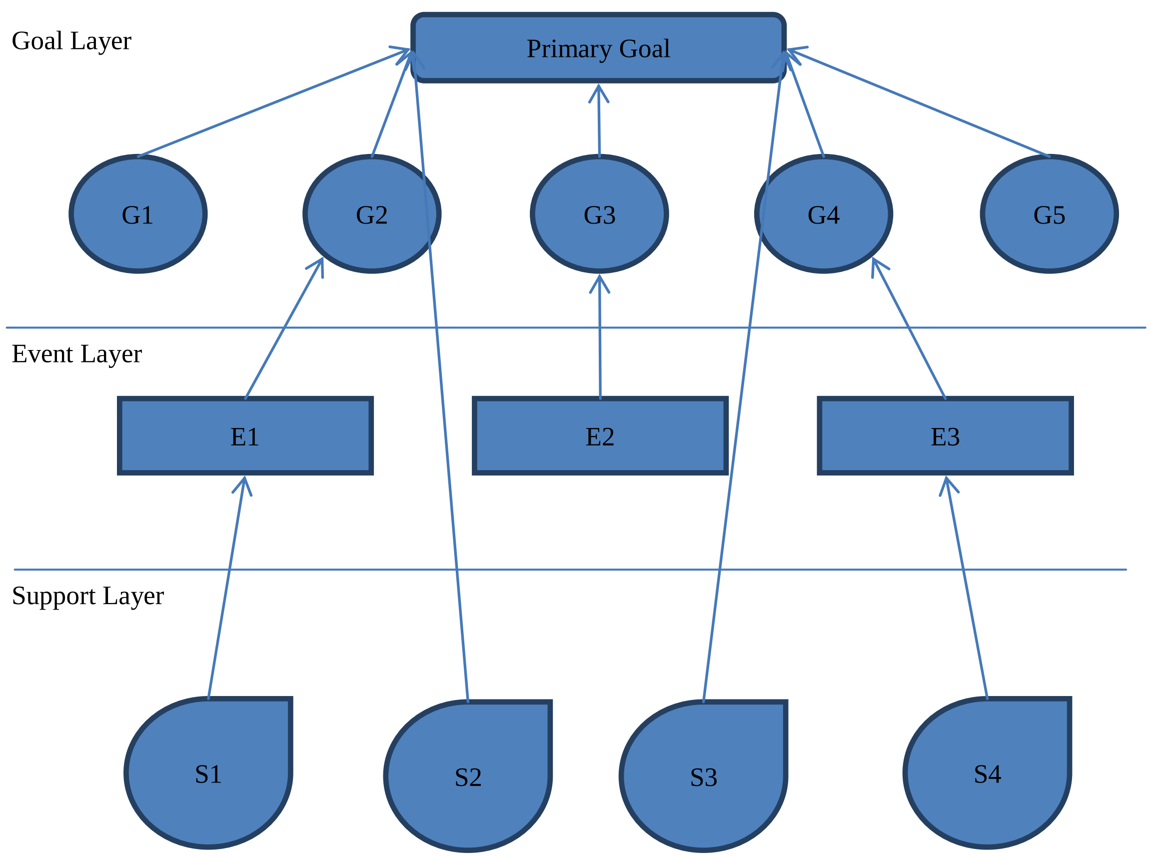


Fig.1. Tropos Goal Model

The relations R is the relations defined over different nodes in the defined goal risk model. The relation can be represented as $R=[N_1,\ldots,N_n \mapsto N]$, where N is the target node and the $N_1,\ldots,N_n$ 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 differ in the semantics. The goal model depicted in the figure 1 projects a main goal, which is associated to a number of associate goals. The affinities of these associate goals are the main criteria behind the success of the main goal. The success rate is projected based on the cost to which the main goal is achieved with an acceptable risk. The usual costs to risk analysis are based on the SAT value and DEN value of the associated goals. In the proposed approach, we define a methodology, which give priority to the associate goals to minimize the cost and tolerate the error to a certain limit. The proposed approach describes the cost to risk analysis through a case study based on the software development company. The following figure

depicts the illustration of the SDC over the Tropos goal model. The figure 2 shows the illustration of the proposed software development company in the model of Tropos goal model. In the figure, it is shown that the top layer of model contains the target goals and associate goals. The event layer contains the associated events, which are needed for the projection of the goals. The supporting events are listed in the support layer as support values. Now a cost to risk estimation is conducted to find the best cost efficient way to calculate the target goal or achieve the goal. The proposed approach describes 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. 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

The above listed are the candidate solution generated for the proposed software development model. These solutions are generated according to the SAT values and DEN values defined by the Tropos goal model. The solutions generated are only considered the parameters likelihood and accepting factors like SAT and DEN, now the proposed approach initiates an optimization phase to extract the most optimized solutions among the generated candidate solutions.

### 3.2 Genetic Algorithm for optimizing the candidate solutions

The genetic algorithm is one of the commonly used optimization algorithm is the data management domain. We make use of the optimization method of GA in the proposed approach to generate the optimized candidate solutions. The candidate solutions obtained from the Tropos goal model are considered as the initial population, which is subjected for the optimization.

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

Initial population of GA

### 3.3 Fitness evaluation

The cost calculation function is conducted as the fitness function the proposed genetic algorithm for candidate solution. 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. The cost evaluation of a candidate solution is described in the following section.

S6- G3 G4 G8 G10 G11 G12,

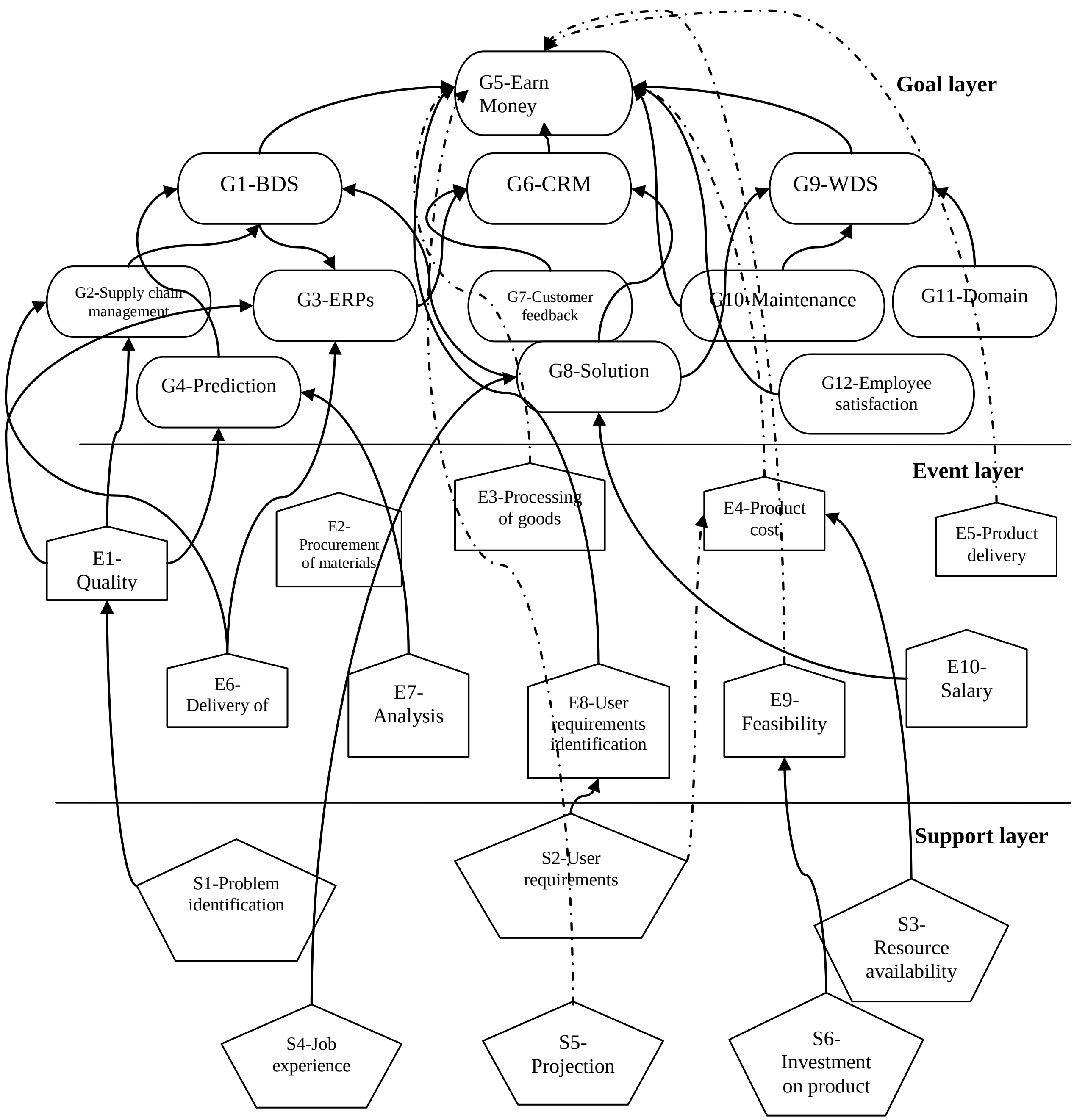


Fig.2. Tropos Goal Model for SDC

Where the associations of different source goals are listed below,

G3➔ SAT (P) E1 (++)→S1 (+)➔E6 (-) = 10+5-5= 10;
G4➔ SAT (F) E1 (++)→S1 (+)➔E6 (+)= 10+5+5=20;
G8➔ SAT (F) E1 (++)→S1 (+)➔E7 (+)= 10+5+5=20;
G10➔ DEN (F) E10 (--)→S4 (-) = 0
G11➔ SAT (P) = 5
G12➔ DEN (P) = 5

COST (S6) = 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. The candidate solutions are then selected for the further genetic algorithm processes. Once the fitness of each candidate solutions are calculated the GA subjects the two other processes such as cross over and mutation

### 3.4 Crossover

The crossover process is one of the characteristic features of the genetic algorithm to improve the stability of the population. The population considered here is the set of candidate solutions obtained after the fitness evaluation. The crossover operation is executed by selecting two candidate solutions from the population and interchanging there characters. A point is set to cross the two candidate solutions and such point is termed as the crossover point. Let us consider the following example; we select the following candidate solutions for the crossover process,

| | | | | | | |
|---|---|---|---|---|---|---|
| G3 | G4 | G8 | | G10 | G11 | G12 |

| | | | | | | |
|---|---|---|---|---|---|---|
| G2 | G3 | G5 | | G6 | G7 | G10 |

Fig.3. parent solutions

In the above figure 3, the two parent candidate solutions are represented, which are subjected for the crossover process. The shaded are in the solutions are considered as the crossover point of the parents.

*Offspring1*

| | | | | | | |
|---|---|---|---|---|---|---|
| G2 | G3 | G5 | | G10 | G11 | G12 |

*Offspring2*

| | | | | | | |
|---|---|---|---|---|---|---|
| G3 | G4 | G8 | | G6 | G7 | G10 |

Fig.4. offspring solutions

The figure 4 represents the offspring solutions obtained after the crossover process and the dark shaded area represent the crossed parent characteristics. The offspring are then subjected for the fitness evaluation.

### 3.5 Mutation

The mutation process is an associated process of the crossover process. The offspring generated after the crossover process are selected for the mutation process. In mutation process, one point is randomly selected from the offspring and a new character is assigned to selected point by replacing the existing one.

*Offspring1*

| | | | | | |
|---|---|---|---|---|---|
| G2 | G3 | G5 | G10 | G11 | G12 |

*New offspring*

| | | | | | |
|---|---|---|---|---|---|
| G2 | G3 | G5 | G10 | G9 | G12 |

Fig.5 mutation

The figure 5 represents the process of mutating an offspring according to the definitions of genetic algorithm. In similar way all the offspring are mutated. According to the genetic algorithm; a new population is created by applying crossover and mutation over the initial population. The feasibility of the newly generated population is calculated based on the fitness evaluation function defined by the proposed approach. Thus after the GA, the resultant will be a set of optimized candidate solutions, which are provided with effective cost and acceptable range of risk.

## 4. Risk Analysis

### 1. Risk prioritization

The cost analysis of the Tropos goal model specifies the cost required for achieving the goals with the association from the event layer and the support layer. The goals are not only associated with cost but also the risk associated with it. The identification of risk is quite a tedious task in the Tropos goal model, because a same element can provide a plus and a negative impact in achieving the goal. So selection of risk should be so specific to achieve the goal with minimum cost and acceptable risk. In the proposed approach, we incorporate a risk prioritization process to analyze the risks. Considering the SAT level and DEN level, the objects in each layer can be grouped to set of risks. But, the level of the risk cannot be identified from those two parameters alone, 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,

$$probabilty(e) = \frac{n(DEN)}{n(DEN + SAT)} \qquad (1)$$

Here, in eq (1) 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. The n(DEN+ SAT) is the number of elements associated with element e having SAT and DEN. According the probability values of the elements, we create a list called risk priority list $R_{list}$.

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

The element is assigned a risk level based on its priority and the SAT level of the elements associated with the element. A threshold is set for characterizing the risk for particular element based on the priority value and SAT level. An element is considered as risk by,

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

So according to the situations, the risk prioritizing factor confirms the elements with most risk for the tropos goal model. The risk prioritizing process helps to differentiate high risk and low risk elements. The elements with higher risk can affect seriously on the cost of the goal, while elements with lower risk level can be mitigated. Thus the risk prioritization helps in reducing the total number of elements, which is considered as risk and that interns speed up the cost analysis phase. Based on the risk values derived from the above expression, we create risk priority list.

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

The $R_{priority\_list}$ is used in the cost to risk analysis phase in order to find the relevant risk to reach the target goals

2. 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 6, 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.

## 5. Experimental Results

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 analyses and those analyses are used to judge the relevant candidate solutions. The experiment uses the input data from a manually generated source as the goal model of Software Development Company.

### 5.1 Case study description and input

The proposed goal risk model uses a case study of a software development company, which is targeted to a prime goal. The prime goal is assigned as "Earn money (G1)" and a number of goals are associated with G1 to make G1 achievable by the SDC. Thus a Tropos goal model is defined over the SDC and by considering the SAT and DEN values a set of candidate solutions are generated. The figure 7(Not Shown) represents the candidate solution generated after the Tropos goal model. We have considered solutions of length 6 to 8 for the efficient evaluation of the cost and risk of the SDC. The associated goals are assigned cost values by randomly generated program and the cost values are associated with the events and supports defined in the Tropos goal model of the SDC. The association list and cost values of different parameters are listed below.

| **Events** | **values** | **Goals** | **costs** | **supports** | **associations** |
|---|---|---|---|---|---|
| E1 | 7 | G1 | 4 | S1 | 8 |
| E2 | 7 | G2 | 5 | S2 | 0 |
| E3 | 4 | G3 | 4 | S3 | 2 |
| E4 | 7 | G4 | 6 | S4 | 4 |
| E5 | 5 | G5 | 1 | S5 | 4 |
| E6 | 0 | G6 | 5 | S6 | 2 |
| E7 | 0 | G7 | 3 | S7 | 7 |
| E8 | 2 | G8 | 6 | S8 | 9 |
| E9 | 4 | G9 | 3 | S9 | 5 |
| E10 | 1 | G10 | 2 | S10 | 5 |

Table.1. value table

Thus, based on the values of the tables, the costs of the generated candidate solutions are calculated. Now, the genetic algorithm is applied to the generated candidate solutions. So, according to the genetic algorithm a number of solutions are extracted from the initial population as the cost effective solutions of the SDC.

| **Candidate solutions** |
|---|
| [G4, G7, G8, G9, G10] |
| [G4, G6, G7, G8, G9] |
| [G2, G7, G8, G9, G10] |
| [ G3, G4, G5, G6, G7] |
| [G2, G3, G4, G5, G6, G7, G8] |
| [G2, G4, G5, G6, G7, G8, G9, G10] |

Table.2. solutions obtained after GA

Now, the cost to risk analysis defined in the proposed approach calculates the cost and relevant risk to the obtained candidate solutions. In the cost to risk analysis phase, the cost and risk of the candidate solution for achieving the target goals are analyses. 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.

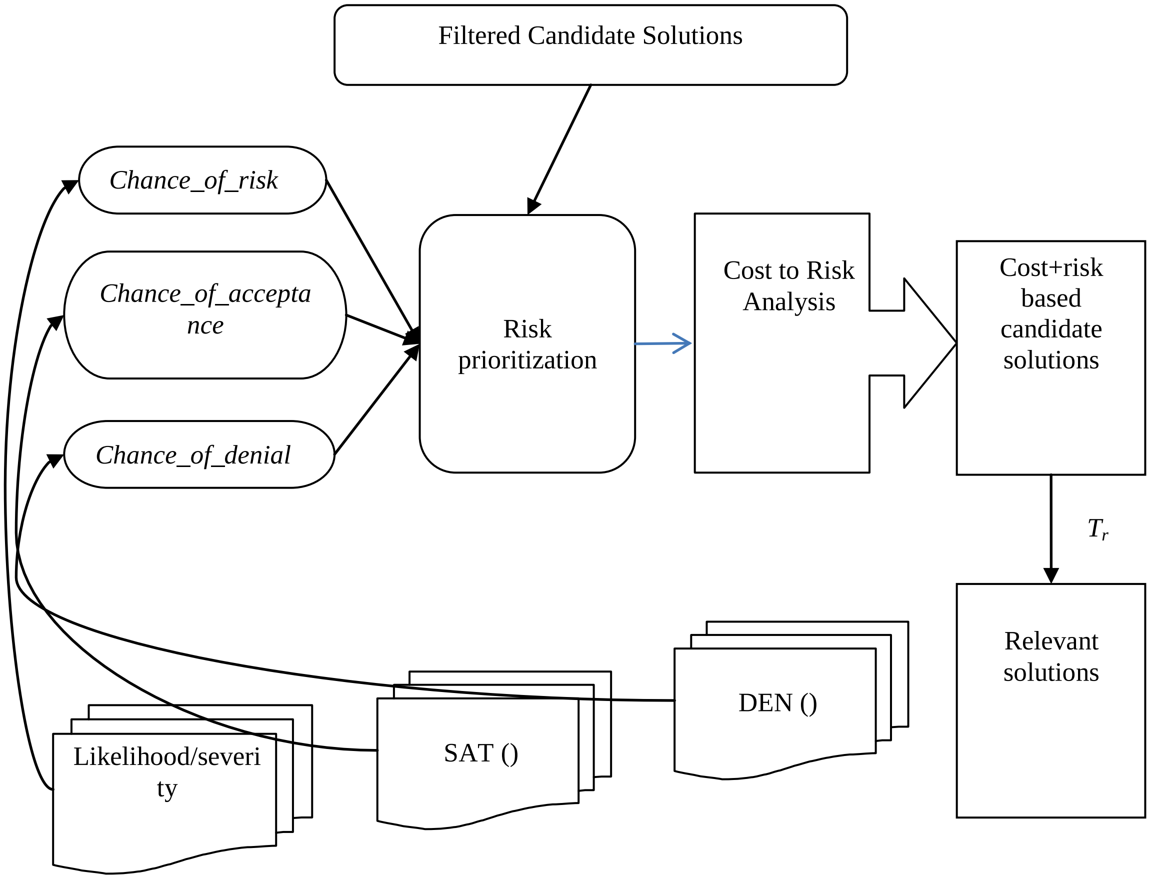


Fig.6. Cost to risk analysis

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 ← 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) ➔ 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. On the cost to risk analysis, we incorporate the risk priority value also with the risk calculation. So, the incorporation of the risk priority value helps in reducing 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 mapped as high and the risk with value 4 is mapped with risk priority low. Thus, according to the priority, we chose risk with value 3 as dominant risk as compared to the risk with Value 4. The cost and risk values of the candidate solutions obtained after GA is plotted in the below table

| Candidate solutions | cost | Risk |
|---|---|---|
| [G4, G7, G8, G9, G10] | 18 | 3 |
| [G4, G6, G7, G8, G9] | 23 | 6 |
| [G2, G7, G8, G9, G10] | 19 | 8 |
| [G3, G4, G5, G6, G7] | 19 | 5 |
| [G2, G3, G4, G5, G6, G7, G8] | 31 | 4 |
| [G2, G4, G5, G6, G7, G8, G9, G10] | 31 | 3 |

Table.3. cost to risk analysis

The table 3 shows the different candidate solutions with cost and risk value and now the task is to select a candidate solution, which cost and risk effective. A risk priority mapping of the calculated risk values are subjected to fix the above problem. A candidate solution is considered after checking with the risk priority graph.

The figure 8 represents the risk priority graph generated from the proposed approach based on the risk values of the candidate solutions obtained after the GA. According to the analysis from the cost to risk table and risk priority graph, the candidate solution “[G2, G3, G4, G5, G6, G7, and G8]” is selected as the effective solution for the SDC. The solution is considered because, it considers most of the associate goals and it has less risk priority as compared to the other candidate solutions

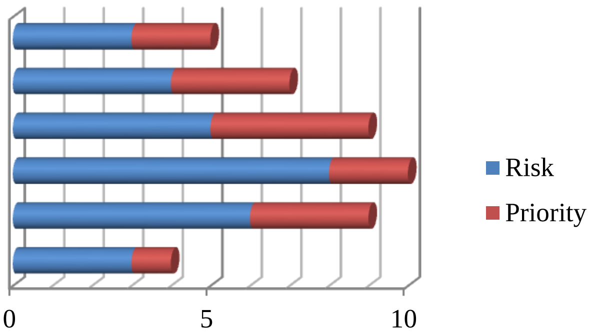


Fig.8. risk priority graph

## 6. Conclusion

The proposed requirement engineering model is 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 proposed approach also adds an optimization technique with the proposed approach. A genetic algorithm driven candidate solution is incorporated with the goal model to get efficient candidate solutions. The risk analysis of the proposed GR model is conducted based on two analyses, the cost analysis, risk priority calculation 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

## 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 YudistiraAsnar Paolo Giorgini March 2007

[5] Focusing on the Importance and the Role of Requirement Engineering Mina Attarha and Nasser ModiriAtta.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 Enginnering (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] JacKyAng, Sook Bing Leong, Chin Fei Lee, UmiKalsomYusof, "Requirement Engineering Techniques in Developing Expert Systems", School of Computer Sciences UniversitiSains 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] YudistiraAsnar, Paolo Giorgini, "Risk Analysis as part of the Requirements Engineering Process", Departmental Technical Report, 2007.

[14] GolnazElahi, Eric Yu, Nicola Zannone, "Security Risk Management by Qualitative Vulnerability Analysis", Third International Workshop on Security Measurements and Metrics, pp: 1-10, 2011.

[15] K. Venkatesh Sharma and Dr P V Kumar "An Efficient Risk Analysis based Risk Priority in Requirement Engineering using Modified Goal Risk Model", International Journal of Computer Applications, Vol. 73, No. 14, pp. 17- 25, 2013

[16] K. Venkatesh Sharma and Dr P. V. Kumar. "An efficient goal oriented risk analysis in requirement engineering." Advance Computing Conference (IACC), 2013 IEEE 3rd International. IEEE, 2013.

[17] K. Venkatesh Sharma, and Dr P. V. Kumar. "An efficient risk analysis in requirement engineering." Engineering (NUiCONE), 2012 Nirma University International Conference on. IEEE, 2012.

**First Author** Mr K. Venkatesh Sharma, is a Research Scholar in JNTU Kakinada, did his B.E (Electronics) from Shivaji University,

Maharashtra, completed M.Tech in Information Technology from Punjabi University, M.Tech in CSE from JNT University, Hyderabad and PGDCA. He has 14 years of teaching experience in various engineering colleges. Currently he is working in CSE dept at TKR College of Engineering& Technology as Assoc Prof.

**Second Author** Dr. P.V Kumar, Professor of CSE in Osmania University Hyderabad, completed M.Tech (CSE) and PhD (CSE) degree from Osmania University, Hyderabad. He has 30 years of Teaching & R&D experience. Many students are working under him for PhD .He has to his credits around 56 papers in various fields of Engineering, which includes Indian ,international journals, National and International conferences, He worked as Chairman BOS in OUCE and conducted various staff development programs and workshops. He is Life Member of ISTE, and CSI societies.