A Qualitative Forward Reasoning Approach for Evaluation of WebGRL Diagrams

Shailey Chawla[1], Sangeeta Srivastava[2], Punam Bedi[3]
[1, 3] Department of Computer Science
[2] Department of Computer Science
Bhaskaracharya College of Applied Sciences
University of Delhi
Delhi, India
shaileychawla@gmail.com, sangeeta.srivastava@gmail.com, pbedi@cs.du.ac.in

ABSTRACT: *At early requirements analysis, resolving conflicts and evaluating design alternatives saves time and effort in the forthcoming phases. Goal oriented approaches are preferred over other requirements engineering approaches for their capability to reason about the requirements and evaluate different solutions. We have enhanced and customized the goal driven reasoning approach for engineering Web applications. Our work is based upon a standard, User Requirements Notation (URN) that has been enhanced by us to suit the Web applications domain and termed WebURN. For different kinds of Web applications and level of domain knowledge, a suitable reasoning approach i.e. Qualitative or Quantitative can be chosen.We have focused on qualitative reasoning methodology in this paper, wherein subjective satisfaction values and contribution values are applied for evaluating the WebGRL graphs. Comparison with other approaches yields that this qualitative approach is more precise, resolves conflicts automatically and enables choice of design alternatives.*

Keywords: Requirements Analysis, Evaluation, Goals, Web Application Engineering, Reasoning, Alternatives, Conflicts

Received: 28 September 2014, Revised 2 November 2014, Accepted 9 November 2014

1. Introduction

According to A. van Lamsweerde (2001) , Goal-oriented requirements engineering (GORE) is concerned with the use of goals for eliciting, elaborating, structuring, specifying, analyzing, negotiating, documenting, and modifying requirements.A goal describes the objectives that the system should achieve through the cooperation of agents in the software-to-be in a given environment as defined by (Liu, L. & Yu, E.2001). Goal oriented requirements engineering employs techniques to clearly understand and capture stakeholder's intentions and motivations. The goal driven models provide good understanding of the system to be developed and express information more clearly. Also, it uncovers the conflicts in early requirements analysis stage, so that they are not propagated further in the software development cycle. It also helps the requirements engineer choose amongst the design alternatives and making decisions amongst various choices. The management of ambiguity and uncertainty at requirements analysis level prevents it from passing on further to the design stage, which would then affect implementation also. This

analytical capability of applying reasoning in early requirements analysis stage makes Goal driven modeling a popular approach.

Goal oriented approaches involve substantial participation of stakeholders, not only for capturing their goals and softgoals but also for analyzing and validating the requirements. Goal models help in thought provocation, and repeated probing over captured requirements so that they are reviewed for further elicitation and also decision making.

Goal models clearly exhibit the dependencies, inter-relationships amongst the intentional elements (Goal, Softgoal, Resource, Task) (A. van Lamsweerde, 2004). The links between these intentional elements enable clear evaluation of different alternatives for implementation. The conflicts are discovered and can be resolved at nascent stages. Different alternatives are analyzed and studied for the impact on other goals and softgoals and the optimized solution can be chosen for further implementation.

The Web application development has been done using the Web Engineering approaches like UWE (Koch, Nora, et al, 2008) , NDT(Jose Escalona, M., and Gustavo Aragón,2008), A-OOH(Garrig´os, I , 2008), WebML(Ceri, Stefano, Piero Fraternali, and Aldo Bongio, 2000) that cater to web specific requirements. However, most of the approaches don't cover the non-functional requirements and even if they do, not so in detail. The non-functional requirementsare very important in software engineering(Chung et al , 2009). There is some work by Lai, Alan, Cui Zhang and Senad Busovaca(2013) on specific non-functional requirements. However, all the non-functional requirements need to be catered. In (Urbieta, M., Escalona, M. J., Luna, E. R., & Rossi, G., 2012), there has been focus on conflict resolution and correcting inconsistencies but it is not goal based. There has also been work by Power, Freire, & Petrie, (2011). Magableh, & Barrett (2011) and Rahman & Meziane, (2011) for designing better web applications but early requirements analysis for creating Web applications has not been covered much in literature. We have provided a goal driven Web Engineering approach in the previous papers (Shailey Chawla, Sangeeta Srivastava, Deepak Malhotra,2013; Chawla, Shailey, and Srivastava, Srivastava, 2012; Shailey Chawla, Sangeeta Srivastava,2012) that incorporates both functional and non-functional requirements and their dependencies in detail. The approach incorporates Goal driven analysis as a core method of requirements engineering for the benefits stated above. In our framework, we have integrated an enhanced User Requirements Notation (I. T. U. T., & Recommendation, Z., 2008), which is a requirements notation standard in the Web requirements Engineering framework. We have also classified both functional and non-functional requirements pertaining to Web applications.

The objective of this paper is to strengthen the analysis of Web Requirements models and enabling better decision making by presenting the reasoning approach for evaluation of Web Goal Requirements Language diagrams. The reasoning of goal model not only evaluates the model for satisfaction analysis but also promotes further thought, analysis, model improvement and increased knowledge about the application domain(Horkoff, J., & Yu, E. , 2011, March). This paper is organized as follows. The next section does an overview of related work in the area of evaluation and reasoning of goal models. In section 3, we explain the overall forward reasoning algorithm that we use in our reasoning approach. Section 4 we have described the Qualitative reasoning methodology that has the set of rules for propagation of satisfaction values from leaf nodes towards the root nodes based on the type of links. In section 5, we apply and relate the developed reasoning method for evaluation of WebGRL diagrams created in our framework using a case study of online book shop. Further, we compare the results of our approach with other prominent approaches in goal based reasoning in Section 6. Lastly we conclude our paper with the inferences and future work directions.

2. Related Work And Motivation

Reasoning methodologies in goal models have been based on various concepts like choice of measurement, propagation method, human intervention and interpretation of goal based syntax. It has been construed that during initial stages of software development and when domain knowledge is limited qualitative reasoning is more appropriate (Jennifer Horkoff, Eric Yu, 2012). The reasoning can be applied either top down (backward) or bottom up (forward) fashion. Forward reasoning answers questions like "*what if this alternative is chosen*?" The leaf nodes are initialized with satisfaction values and through various links, satisfaction values are propagated upwards to infer whether the parent goals would be satisfied and how much satisfaction would be there? In backward reasoning approaches parent goal's satisfaction value is specified and it is propagated through various links and goals to various leaf nodes. It answers the question "*how the parent goal can achieve the satisfaction*?"

Horkoff and Eric Yu have also given both forward and backward reasoning approaches (Horkoff, J., & Yu, E., 2009; Horkoff, J., & Yu, E., 2010). In paper (Giorgini, P., Mylopoulos, J., & Sebastiani, R., 2005) Tropos methodology gives both forward and backward reasoning approach. The Tropos methodology has been formalized in (Paolo Giorgini, John Mylopoulos, Eleonora

Nicchiarelli, Roberto Sebastian, 2003) where axiomitization has been done for all the rules. The uncertainties and incompleteness in partial goal models has been handled in (Salay, R., Chechik, M., & Horkoff, J., 2012) so that they are not passed off to design and implementation phases. There are other approaches also like the ones based on constraints(Luo, Hao, and Daniel Amyot, 2011) or mixed(Roy JF, 2007) or metrics based approaches(Franch X, 2006);Kaiya H, Horai H, Saeki M , 2002) . The work in (Antoine Cailliau and Axel van Lamsweerde, 2012) incorporates a probabilistic approach for goal model analysis for the KAOS model. There is detailed survey and analysis of most of the goal reasoning approaches in (Jennifer Horkoff, Eric Yu, 2012) and also an analysis of what approach suits a particular situation is given in (Horkoff, J., & Yu, E. , 2011, March).In (van Lamsweerde, A., 2009; A. van Lamsweerde, 2009) another analysis methodology has been developed for reasoning and choice of alternatives in goal models. Requirements analysis also involves the role of priorities or preferences of stakeholders. In (Liaskos, S., McIlraith, S. A., Sohrabi, S., & Mylopoulos, J., 2010, September) the preferences of different stakeholders for particular requirements have been integrated to goal model analysis.

A decent body of work has been done for reasoning and evaluation of goal models. Most of the work focuses on a main area like conflict resolutions, alternative selection or preference of requirements. However, it has been observed by us that the treatment of links doesn't take into account priority of the goals. The qualitative approaches give a very vague idea of the goal accomplishment and reasoning of alternatives. Many algorithms described above do not resolve the conflicts automatically and human intervention is required. Hence, in our paper we develop a qualitative approach with increased precision, conflict detection and resolution and enable choice of alternatives. Tool support for automating these processes has also been done. For conflict resolution, we have employed the priority of goals to come to a solution.

3. Forward Reasoning Algorithm

A forward reasoning algorithm starts with an analysis question "what if a particular alternative is chosen?" In this algorithm intentional element refer to goals, softgoals and tasks. Resource is not considered for evaluating the satisfaction values because their role is mainly to provide or store the information guided by the other intentional elements. Any links to or from the Resources are also ignored. The procedure initializes some intentional elements primarily the leaf goals and propagates upwards to reach the primary goals. It makes use of evaluation rules and labels to express degree of satisfaction or denial. The evaluation rules vary with the type of links being used in the goal graph.

The forward reasoning algorithm consists of the following steps:

1. Decide the type of evaluation to be used. (Qualitative or Quantitative)

2. Initialize the Satisfaction values for leaf nodes (Goals, Softgoals and Tasks)

3. Create two lists –

a. **ElementsReady:** Intentional elements whose satisfaction values have been initialized/calculated.

b. **ElementsWaiting**:- Intentional elements whose satisfaction values can't be calculated yet.

4. Calculate the satisfaction values of elements in waiting list using the elements in Ready list using the following order of the links:

a. Decomposition links

b. Means End links

c. Dependencies

d. Contribution links

5. After the evaluations of all the intentional elements have been done, the algorithm can be re-evaluated for alternative choices of design.

6. Different evaluations are then compared for choosing the best outcome/ alternative.

Our work is based on different reasoning approaches for Goal oriented Requirements Engineering as discussed in Section 2. We have enhanced the work on reasoning approaches in various ways. Our contributions are listed below:

• The forward reasoning algorithms in all the previous approaches differ in treatment of AND decomposition. They propagate the minimum satisfaction value to the parent goal. We, however, propose using weighted average for computing AND decompositions because we trust choosing the minimum satisfaction values propagates inappropriate satisfaction values.

• We have added more detail to the qualitative model, by adding another intermediate satisfaction value i.e. weakly satisfied (WS). This value falls between unknown satisfaction (UN) and partial satisfaction (PS). So instead of 5 levels, we have 7 levels of satisfaction.

• Resolve conflicts automatically, using the priority of goals/ softgoals.

• Integrate the algorithm to our framework, which deals with different levels of refinement. The Common goals / softgoals that exist in two or more diagrams are dealt with.

The activity diagram depicting the steps to be undertaken for evaluating WebGRL diagram is shown in figure 1.

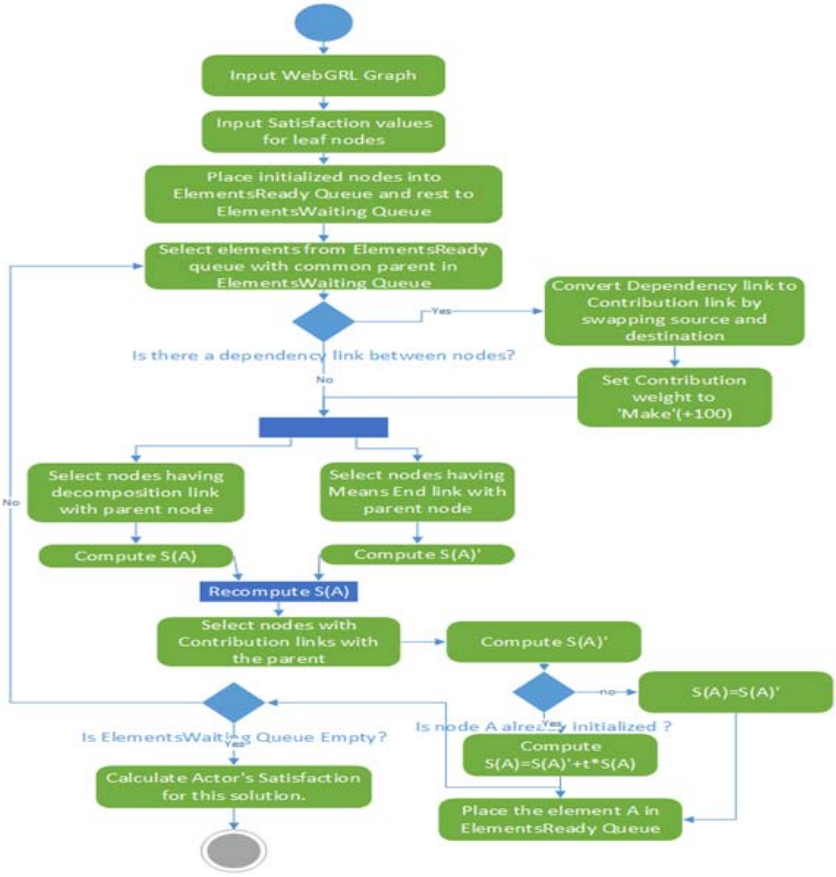

Figure 1. Activity Diagram depicting the Evaluation of WebGRL Diagram

The following terminology would be used throughout this paper for evaluation of WebGRL diagrams:

A is a parent goal

***Xi* 's** are subgoals $i = 1$ to n that have some links attached with *A*.

***S* (*A*)** is the total satisfaction of goal / softgoal *A*.

***S* (*A*, *Xi*)** is the satisfaction of goal/softgoal *A* wrt *Xi*.

***C* (*A*, *Xi*)** is the contribution weight of goal *Xi* towards goal *A*.

***P* (*Xi*)** is the priority of goal/softgoal *Xi*.

4. Qualitative Forward Reasoning Algorithm (FREQ)

A qualitative approach signifies that instead of using the exact numeric values, qualitative labels are used to specify values of variables. The qualitative approach is easier to understand and apply at early stages of requirements analysis as the requirements model doesn't give a very crisp representation at this point. We follow the forward reasoning or bottom up approach that calculates the satisfaction values of parent nodes/ root nodes from the leaf or intermediate nodes. The algorithm is executed for different alternativesthe algorithm used here is called **FREQ** (Forward Reasoning and Evaluation –Qualitative) algorithm.

For qualitative analysis and evaluation the following subjective labels can be used for initializing the intentional elements.

- The priority of goal/ softgoal *Xi* is given by *P* (*Xi*) = *enum* {*None, Little, Moderate, Important, Critical*}, such that Critical > Important > Moderate > Little > None.
- The satisfaction of a goal/softgoal *Xi* is given by *S* (*Xi*) = *enum*{*Fully Satisfied* (*FS*), *Partially Satisfied*(*PS*), *Weakly Satisfied(WS), Unknown* (*UN*), *Weakly Denied* (*WD*), *Partially Denied* (*PD*), *Fully Denied* (*FD*)}such that FS > PS > WS > UN > WD > PD > FD.
- The qualitative contribution representations are also represented in similar fashion. The contribution of a goal/softgoal *Xi* towards satisfaction of goal/softgoal A is represented as follows: *C* (*A, Xi*) = *enum* {*Make, Help, Some +, Unknown, Some -, Hurt, Break*}

The values in the set are represented such that Make is maximum contribution towards satisfaction of parent goal and Break is maximum contribution towards Denial, that is, negative contribution. Likewise, Help and Hurt have little lesser degree of contribution towards Satisfaction and Denial respectively. Similarly for Some+ and Some- corresponding to satisfaction and denial. Unknown means contribution is not known and can be understood as numeric zero. For reference with the quantitative contribution methods, these can be understood as shown below in Figure2. We assume that values lie between -100 and +100, wherein -100 to -90 mean fully denied, -10 to +10 means unknown, +90 to +100 mean fully satisfied.

A goal can be satisfied or denied or the value can be unknown as abbreviated; S-Satisfaction, D-Denial, UN-Unknown. The degree of satisfaction or denial can be F- full, P-Partial, W-Weak. When one subgoal is denied and other satisfied there can be a conflict. It is denoted as C; conflict can be of three types S-Strong, M-Medium and W- Weak.

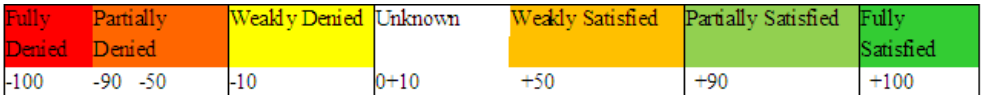

Figure 2. Satisfaction values (Qualitative and Quantitative models)

4.1 Rules For Calculating Satisfaction Of Parent Goal Wrt Different Links

The satisfaction value of parent goal from a set of child nodes is evaluated based on the link that connects the child with the parent node. There are four kinds of links supported in WebGRL notation- Decomposition (AND/OR), Dependency, Means-end and Contribution links.

4.1.1 AND Decomposition

The Decomposition of a goal defines what is needed for a fulfillment of the goal, mainly refinement. When a goal A is decomposed into two or more subgoals with an AND decomposition link then all the subgoals should be satisfied for satisfaction of parent goal A. The approach for calculating the satisfaction of *A* with a AND decomposition has been proposed by (Horkoff, J., & Yu, *E*., 2009; Paolo Giorgini, John Mylopoulos, Eleonora Nicchiarelli, Roberto Sebastian, 2003;Daniel Amyot et al, 2010). They suggest that minimum satisfaction value amongst the subgoals should be propagated. However, according to our understanding we suggest and propose that average value should be propagated to the parent goal in case of AND decomposition. If the subgoals are partly satisfied then it would be wrong to say that parent goal is not satisfied at all. We shall explain this with the help of a quantitative example. If *X*1,*X*2 and X3 are the subgoals and their satisfaction values are +100, +80 and +20 respectively, then A's satisfaction should be the average, i.e. +66.6. That is, if *X*1 is fully satisfied, *X*2 ispartially satisfied and *X*3 is weakly satisfied then, it means that A is around 2/3rd satisfied. The previous proposals suggest that minimum value should be propagated, which would mean the satisfaction of A would be only +20. However, in this example, if two of three subtasks are fully satisfied, then A should have been ideally 2/3rd satisfied, and minimum satisfaction of +20 doesn't do justice to satisfaction of A.

Therefore, we propose an average satisfaction of subgoals as per Table 1,should be taken for calculating parent goal's satisfaction. Figure 3 depicts an example of nodes connected with AND decomposition. The resultant satisfaction value of goal A in this case would be PS as per table 1.

	FD	PD	WD	UN	WS	PS	FS
FD	FD	PD	PD	WD	MC	MC	SC
PD	PD	WD	WD	WD	WC	MC	MC
WD	WD	WD	UN	UN	UN	WC	MC
UN	PD	WD	UN	UN	UN	WS	PS
WS	MC	WC	UN	UN	UN	WS	WS
PS	MC	MC	WC	WS	WS	WS	WS
FS	SC	MC	MC	WS	PS	PS	FS

Table 1. Resultant Satisfaction values from two nodes

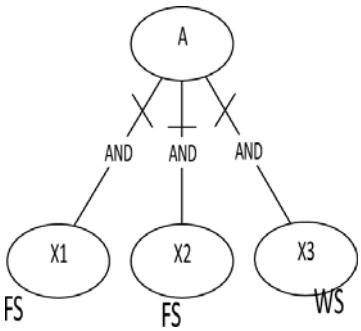

Figure 3. Calculating satisfaction value of parent from nodes linked with AND decomposition

4.1.2 OR Decomposition Link

OR decomposition link signifies that a parent goal is decomposed into two or more subgoals such that completion or satisfaction of any one of them would suffice the satisfaction of the parent goal. In case of OR decomposition the maximum satisfaction value is propagated to the parent goal. For a given parent goal *A* and its subgoals in OR decomposition *X*1……*Xn*, the satisfaction of *A*, would be given as

$$S(A) = max_{i=1\ to\ n}\ S(Xi)\ \ s.t.\ n > = 2$$

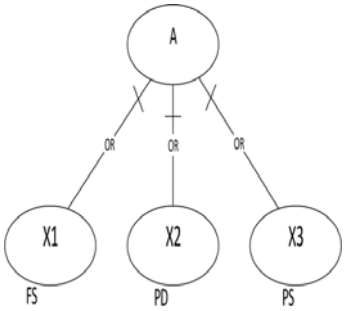

Figure 4. Calculating satisfaction value of parent from nodes linked with OR decomposition

In the example in Figure 4 above, since *S* (*X*1) has the maximum satisfaction value, Fully Satisfied, hence, FS would be propagated to A.

4.1.3 Means End Links

The means end links signify that for actualization or operationalization of goals, there have to be tasks that act upon them, Here, *S* (*Xi*) can be either denied or satisfied. This is because in a means end relationship, the goal or softgoal is related with tasks, *X*1,…., *Xn* that would either be completed or not completed. For deriving the satisfaction value of goal *A*, table 1 is used similar AND decomposition links.

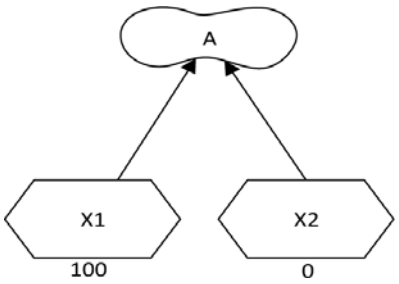

Figure 5. Satisfaction value from nodes linked with means end links

In the above example in figure 5, *X*1 has satisfaction value *FS* and *X*2 has satisfaction value FD. Hence the satisfaction of A would be UN, unknown.

In case of just one task, the same value is propagated towards, the parent goal.

4.1.4 Dependency Link

The dependency link as shown in figure 6 depicts that satisfaction of goal *A* is dependent on satisfaction of softgoal *B*. Similar to work of (Jennifer Horkoff, Eric Yu, 2012) we would transform the dependency link to a contribution link in the opposite direction. So, if *A* is dependent on *B* for its satisfaction we say that *B* makes *A*. So a contribution link with source *B* and target *A* is created with the weight "*Make*".

Figure 6. Calculating satisfaction value of a node connected with dependency link

4.1.5 Contribution Link

The contribution link from source subgoal to a destination parent goal specifies, how the satisfaction of the source goal affects the satisfaction of the parent goal. The contribution can be either positive or negative. Numerically the values range from -100 to +100. Correspondingly, the qualitative values range from Fully Denied to Fully Satisfied as explained above. A particular node can have multiple incoming contribution links, in such cases they are undertaken collectively. If *A* is destination node / parent goal and *X*1……*X*n are the source nodes/ contributing goals/softgoal, then *S* (*A*, *Xi*) represents the contribution of goal *Xi* towards satisfaction of A and *S* (A) signifies total satisfaction of *A*. *S* (*A*) is derived after combining the values of *S* (*A*, *Xi*) for i = 1 to *n*. Table 2 below specifies the calculation of *S* (*A*, *X*i) based on the satisfaction of *Xi* and *C* (*A*,*Xi*), contribution weight of the link connecting *A*, *Xi*. The combined satisfaction is calculated by using the Table 2 given below in pairs, rather than all at once. Like for n values of *S* (*A*, Xi), first *S* (*A*, *X*1) and *S* (*A*, *X*2) are combined, then *S* (*A*, *Xt*) and *S* (*A*, *X*3) and so on.

	Make	Help	Some+	Unknown	Some-	Hurt	Break
FD	FD	PD	WD	UN	WS	PS	FS
PD	PD	WD	WD	UN	WS	WS	PS
WD	WD	WD	UN	UN	UN	WS	WS
UN	UN	UN	UN	UN	UN	UN	UN
WS	WS	WS	UN	UN	UN	WD	WD
PS	PS	WS	WS	UN	WD	WD	PD
FS	FS	PS	WS	UN	WD	PD	FD

Table 2. Satisfaction value of single node based on its satisfaction value and contribution weight

For example, in Figure 7, *X*1, *X*2 and *X*3 contribute towards satisfaction of goal *A*. When combining the satisfaction values of different goals, firstly, all positive or satisfaction values are combined together. Next, all negative or denial values are combined together. Thereafter, satisfaction and denial values are combined accordingly. According to Table2, *S* (*A*, *X*1)would be *PS*, *S* (*A*, *X*2) would be *WD* and *S* (*A*, *X*3) would be *WS*. On combining *S* (*A*, *X*1) and *S* (*A*, *X*3) we get WS. On combining *WS* with *S* (*A*, *X*2), the result is unknown contribution, which is propagated to node *A*.

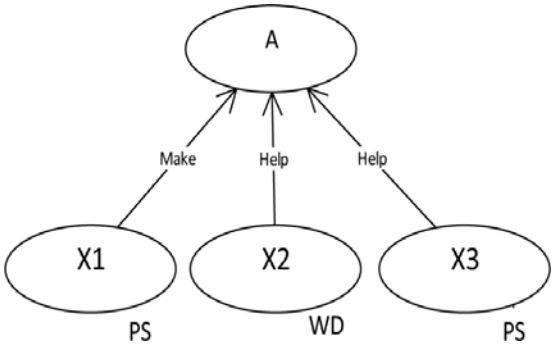

Figure 7. Calculating Satisfaction value of parent based on nodes linked by contribution link

4.1.6 Dealing with Conflicts

If a goal or softgoal has incoming positive and negative contribution then there can be a potential conflict among requirements. In case of AND decomposition relations also, positive and negative satisfaction values i.e. Denial and Satisfaction values together can cause conflicts. For resolving the conflicts, it is suggested that priority of the goal/ softgoal should be considered to reach the combined satisfaction value. As suggested above the type of conflict can also be stated depending on the values of individual satisfaction values. The type of conflict as shown in Table 3, is reported to the requirements engineer by comparing the values of *S* (*A*, *Xi*) for all the subgoals.

	FD	PD	WD
FS	Strong	Medium	Medium
PS	Medium	Medium	Weak
WS	Medium	Weak	Weak

Table 3. Types of Conflict

The satisfaction of goal *Xi*, *S* (*A*, *Xi*) is altered using its priority value.

- For priority values, Indispensable and Critical, it remains as it is.
- For priority values, important and moderate, the satisfaction is brought down one degree.
- For little or not important goals, the satisfaction value is brought down two degrees.

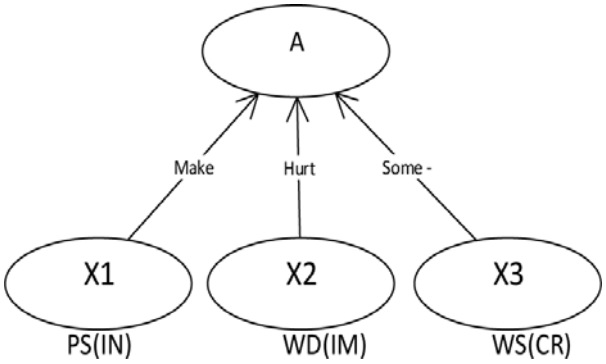

Figure 8. Resolving conflicts using priorities

By degrees, it means, the intensity of satisfaction or denial. Full becomes partial or weak. However, a weak satisfaction wouldn't turn into weak denial. In such a case, it is kept as unknown. The newly computed satisfaction values *S* (*A, Xi*) are then combined together such that

$$FS \wedge FD \rightarrow UN, PS \wedge PD \rightarrow UN, WS \wedge WD \rightarrow UN$$

$$FS \wedge PD \rightarrow WS, PS \wedge FD \rightarrow WD$$

$$FS \wedge WD \rightarrow PS, FD \wedge WS \rightarrow PD$$

$$PS \wedge WD \rightarrow WS, WS \wedge PD \rightarrow WD$$

This leads to more appropriate satisfaction values as priority of a goal/softgoal is also considered besides the satisfaction value of the goal. The conflicts are resolved by incorporating the importance or priority of the goals. A goal with more satisfaction but very less priority becomes less satisfied so that a goal with a higher priority value can contribute more towards satisfaction of the parent goal. The example in figure 8 above depicts resolving conflicts by using the priority values of the nodes. The satisfaction value is recomputed and then the resultant satisfaction value is re-calculated. After this calculation the element is added to the ElementsReady Queue and satisfaction values of other elements are calculated in the same order of links. After all the elements have been added to ElementsReady Queue, the total satisfaction value is calculated for the Actor for the current solution.

4.1.7 Actors Satisfaction

The Actor's satisfaction is calculated for a particular solution by incorporating weighted priority of a goal or softgoal according to the Algorithm given by Daniel Amyot et al (2010). The enhancement to the Algorithm is in the levels of satisfaction that have been increased to 7 from 5 levels. The enhanced algorithm that calculates the Actor's satisfaction is given below

5. Applying Reasoning To Web Requirements Engineering Framework

In the web requirements engineering framework, proposed in (Chawla, Shailey, and Srivastava, Srivastava, 2012; Shailey Chawla, Sangeeta Srivastava, 2012), the standard of User Requirements Notation has been enhanced for Web specific functional and non-functional requirements. The approach elicits initial goals from the stakeholders first, and generates a Base WebGRL diagram to depict those goals. The framework provides process support for guiding the requirements engineer to cover all the steps for creating the base WebGRL diagram. After validating the Base WebGRL diagram for any errors and discrepancies, there is further elicitation for Web specific functional goals- Content, Navigation, Business Process, Presentation and Adaptation. For each web specific functionality,a specific WebGRL diagram is created that focus on the functionality in detail covering its functional and non-functional requirements. There is constant support provided for the flow of the requirements engineering process and each diagram is validated.

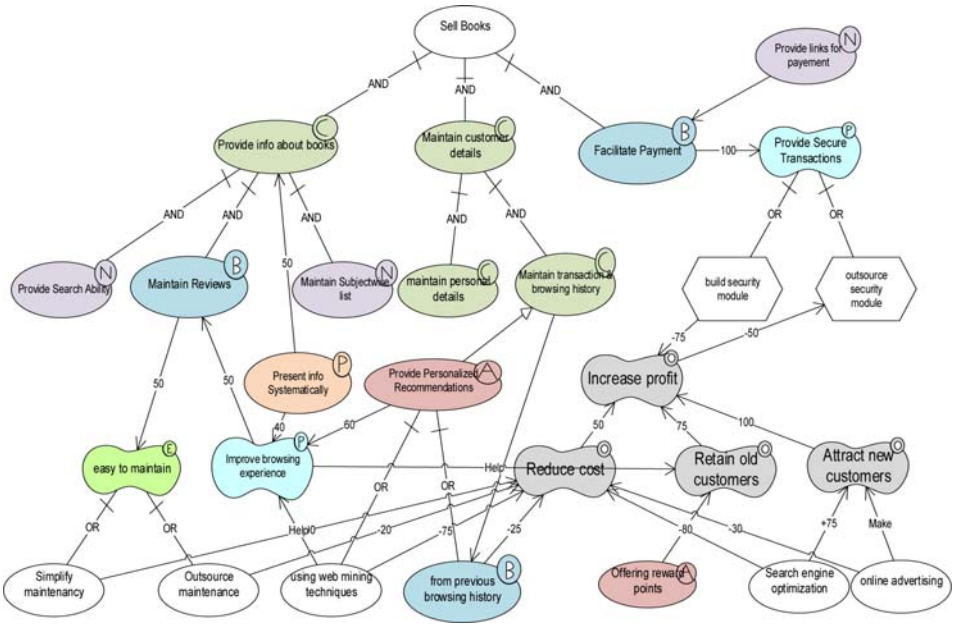

Figure 9. WebGRL diagram for Online Bookshop

```
Algorithm CalculateActorEvaluation
Inputs WebGRL Model
Output actorEvalValue:QualitativeLabel
oneElemVal:QualitativeLabel // 1 elem. value weighted according to its impor-
tance
ns:Integer = 0 // number of Satisfied weighted values
nws:Integer = 0 // number of WeaklySatisfied weighted values
nwd:Integer = 0 // number of WeaklyDenied weighted values
nps:Integer = 0 // number of PartiallySatisfied weighted values
npd:Integer = 0 // number of Partially Denied weighted values
nd:Integer = 0 // number of Denied weighted values
nu:Integer = 0 // number of Undecided weighted values
weightSD:QualitativeLabel // partial weighted values from ns and nd
weightWSWD:QualitativeLabel // partial weighted values from nws and nwd
weightPSPD:QualitativeLabel // partial weighted values from nps and npd
// compute the numbers of weighted contributions for each kind
for each boundElem:IntentionalElement in actor.elems
{
oneElemVal = WeightedImportance(boundElem.qualitativeVal,
boundElem.importance)
AdjustEvaluationCounters(oneElemVal, ns, nws, nwd, nd, nu, nc)
}

// check for the presence of undecided and conflict weighted evaluation values
if (nc > 0)
actorEvalValue = Conflict
else if (nu > 0)
actorEvalValue = Undecided
else
{
weightSD = CompareSatisfiedAndDenied (ns, nd)
weightWSWD = CompareWSandWD (nws, nwd)
weightPSPD = ComparePSandPD (nws, nwd)
actorEvalValue = CombineContributions (weightSD, weightWSWD, weightPSPD)
}
return actorEvalValue
```

In forward reasoning approach, for a given goal model, the leaf nodes are initialized with the satisfaction values and for a particular set of nodes it reasons if the root goal along with other subgoals are fulfilled by means of a set of rules. The leaves propagate the satisfaction values upwards based on a set of rules and satisfaction level of all the other goals is found.

For evaluation, the leaf nodes are initialized. The leaf nodes here have special meaning pertaining to different types of links. In WebGRL diagram, the leaf nodes are with following conditions:-

No *outgoing**decomposition or dependency***links.

No *incoming* ***contribution links or means-end*** links.

There are two lists maintained for initialized and non-initialized elements: elementsReady and elementsWaiting respectively. The diagrams are evaluated with the rules as described in the flow chart in the previous section till there are no elements in elementsWaiting list. We take the example of WebGRL diagram shown in Figure 9 to exhibit our reasoning approach. First of all the leaf nodes are identified and initialized according to Table 4.

Leaf node	**Initial Satisfaction**
Maintain reviews	FS
Provide search ability	FS
Maintain subjectwise list	FS
Maintain Personal information	FS
Maintain transaction & browsing history	FS
Online advertising	FS
Search engine optimization	PS
Offering reward points	FS
Present info systematically	FS
Outsource	FS
Simplify maintenance	PS
Build security module	PS
Outsource security	FS
Based on previous browsing history	FS
Using web mining	FS

Table 4. Initialization of leaf nodes for WebGRL diagram given in Figure 9

6. Comparison of FREQ with Other Algorithms

We have compared our algorithm with the prominent algorithms (Daniel Amyot et al, 2010) in goal model reasoning for evaluation of GRL diagrams. We have applied the algorithms for evaluation of the WebGRL diagram of an Online BookShop. The primary objective is to sell books and increase profit. A Base WebGRL diagram has been created by us using Microsoft Visio 2013 capturing the objectives and recreated in the other tool i.e. jucmNav for analysis. The leaf nodes in the WebGRL diagram had been identified and initialized as shown in Table 4. We choose two alternatives for comparing the satisfaction values.

Alternative 1: Drop the intentional elements: Simplify maintenance, Build security module and Based on previous browsing history.

Alternative 2: Drop the intentional elements: Outsource maintenance, Outsource Security and Use Web mining Techniques.

Since our work is based on User requirements notation, we have compared with six algorithms applied on URN and implementedin jUCMNav (Mussbacher, Gunter, and Daniel Amyot, 2009). The algorithms are Qualitative algorithm, Quantitative algorithm and mixed algorithm, constraint solver and conditional algorithm (Daniel Amyot et al, 2010; Jennifer Horkoff .2013). It is noticed that on applying various algorithms the qualitative results are almost the same in all the algorithms mentioned in (Daniel Amyot et al, 2010; Jennifer Horkoff .2013). Though the numeric values differed but because of lack of qualitative segments they were put into weakly satisfied or denied area. For qualitative analysis only Full satisfaction/denial, Weak satisfaction denial and unknown values are there. However, if we consider the quantitative range of -100 to +100 the satisfaction levels nearing 0 or 100 can't be

all put in weak satisfaction segment. Hence, in our approach we have added another segment called partial satisfaction that has the values between 50 and 90. The weak satisfaction value ranges between 10 to 50. Algorithm shows statistically different results and also, more varied because now we have seven qualitative parameters to interpret the satisfaction level of goals or softgoals. Also the strategy is different from previous approaches in treatment and sequence of the links to a single parent goal. Table 5 shows comparison of results of satisfaction values for different goals using different algorithms.

		ALTERNATIVE1							ALTERNATIVE2						
Intentional Elements	Priority	Qualitative	Quantitative	Mixed	Constraint	Formula	Conditional	FREQ	Qualitative	Quantitative	Mixed	Constraint	Formula	Conditional	FREQ
Sell books	H	FS	100	FS	FS	FS	FS	FS	FS	90	90	FS	FS	FS	FS
Facilitate Payments	H	FS	100	FS	FS	FS	FS	FS	FS	75	75	FS	FS	FS	FS
Provide info about books	H	FS	100	FS	FS	FS	FS	PS	FS	100	100	FS	FS	FS	PS
Maintain customer details	H	FS	100	FS	FS	FS	FS	FS	FS	100	100	FS	FS	FS	FS
Reduce cost	M	WD	-73	FD	FD	FD	FD	FD	FD	-100	-100	FD	FD	FD	WD
Increase profit	H	WS	90	WS	WS	WS	WS	WS	WS	29	44	WS	WS	WS	PS
Retain old customers	M	WS	90	WS	WS	WS	WS	WS	WS	90	90	WS	WS	WS	WS
Attract new customers	H	WS	43	WS	WS	WS	WS	WS	WS	43	57	WS	WS	WS	WS
Easy to maintain	L	FS	100	FS	FS	FS	FS	FS	FS	90	90	FS	FS	FS	FS
Improve browsing experience	II	WS	90	WS	WS	WS	WS	WS	WS	90	90	WS	WS	WS	PS
Provide secure transactions	II	FS	100	FS	FS	FS	FS	PS	WS	75	75	FS	FS	FS	PS
provide personalized recommendations	M	FS	100	FS	FS	FS	FS	FS	FS	FS	100	FS	FS	FS	FS
Maintain reviews	M	FS	100	FS	FS	FS	FS	FS	FS	100	100	FS	FS	FS	FS
Provide search ability	H	FS	100	FS	FS	FS	FS	FS	FS	100	100	FS	FS	FS	FS
Maintain subjectwise list	II	FS	100	FS	FS	FS	FS	FS	FS	100	100	FS	FS	FS	FS
Maintain Personal information	M	FS	100	FS	FS	FS	FS	FS	FS	100	100	FS	FS	FS	FS
Maintain transaction & browsing history	M	FS	100	FS	FS	FS	FS	FS	FS	100	100	FS	FS	FS	FS
Online advertising	H	FS	75	FS	FS	FS	FS	FS	FS	75	100	FS	FS	FS	FS
Search engine optimization	H	WS	50	WS	WS	WS	WS	WS	WS	50	50	WS	WS	WS	PS
offering reward points	M	FS	100	FS	FS	FS	FS	FS	FS	100	100	FS	FS	FS	FS
present info systematically	M	WS	75	WS	WS	WS	WS	WS	WS	75	75	WS	WS	WS	PS
Outsource	L	FS	100	FS	FS	FS	FS	FS	X	X	X	X	X	X	X
simplify maintenance	L	WS	X	X	X	X	X	X	WS	75	75	75	75	WS	PS
Build security module	.....	X	X	X	X	X	X	X	WS	75	75	WS	WS	WS	PS
Outsource security	.....	FS	100	FS	FS	FS	FS	FS	X	X	X	X	X	X	X
based on previous browsing history	.....	FS	X	X	X	X	X	X	FS	100	100	FS	FS	FS	FS
using web mining	.....	FS	100	FS	FS	FS	FS	FS	X	X	X	X	X	X	X

Table 5. Comparison of results of various algorithms for two alternatives (refer figure 9)

The satisfaction values are computed for two different alternatives. In the first alternative, the goals/ tasks simplify maintenance, build security module and based on previous browsing history are omitted. All the three intentional elements mentioned here were in OR decomposition relationship with the corresponding parent goal. For alternative 2, the other bunch of intentional elements i.e. outsource maintenance, outsource security and use web mining techniques are omitted. The satisfaction values are computed for both alternatives by various algorithms. The results differ slightly for various algorithms and in our algorithm FREQ, the results vary more because of larger set of values that can be taken up for satisfaction of an intentional element. The Actor's Satisfaction is PS for Alternative 2 and WS for Alternative 1. This implies FREQ Algorithm favours Alternative 2 over Alternative 1. Figure 10 below shows the graph depicting the range of satisfaction values of different goals/softgoals corresponding to Base WebGRL diagram in Figure 9 for two different alternatives. The total/average satisfaction determines what alternative is to be chosen.

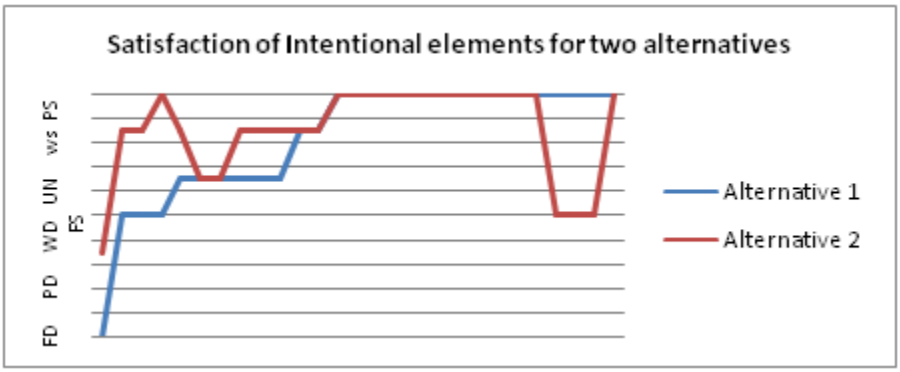

Figure 10. Satisfaction values for two alternatives for evaluating Base WebGRL diagram

The alternatives suggested by other algorithms are shown in Table 6 below. Most of the algorithms suggest Alternative 2 over Alternative 1 except Qualitative model for GRL and Conditional Algorithm for GRL. FREQ Algorithm also suggests choice of Alternative 2 i.e. dropping the outsource security and maintenance and also not using Web mining for the design solution. This receives maximum satisfaction keeping in mind the priority of the goals provided by the stakeholders.

Algorithm	Alternative chosen
Qualitative, GRL	1
Quantitative, GRL	2
Mixed	2
Hybrid	2
Conditional	1
Formula Based	2
FREQ	2

Table 6. Alternatives chosen by various Algorithms

7. Conclusion

In this paper, we have presented a qualitative approach to reason about the goals and evaluate the requirements model created for provoking further discussion and improved understanding. The forward reasoning approach works in a bottom up fashion wherein the leaf nodes are initialized and the satisfaction values are propagated towards the root or primary goals. We differ in the forward reasoning approach from other goal based methodologies as; we pay more emphasis on Decomposition and Means-end links than Dependency and contribution links. We enable proper visualization, analysis and resolution of conflicts as our approach detects and resolves the conflicts based on the priorities of respective goals. The computation of satisfaction values for goals for certain link categories has been altered from previous approaches for more appropriate results. Tobring more precision to qualitative assessment, the array of satisfaction values has been augmented. The evaluation enables the requirements engineer to choose the design option amongst a set of alternatives, besides resolving the conflicts. The end product is a requirements model free of conflicts and proposal of a better design alternative.

References

[1] Amyot, D., Ghanavati, S., Horkoff, J., Mussbacher, G., Peyton, L., Yu, E. (2010). Evaluating goal models within the goal oriented requirement language. *International Journal of Intelligent Systems*, 25 (8) 841-877.

[2] Cailliau, A., & van Lamsweerde, A. (2012, September). A probabilistic framework for goal-oriented risk analysis. *In: Requirements Engineering Conference (RE)*, 2012 20th *IEEE International* (p. 201-210). IEEE.

[3] Ceri, S., Fraternali, P. Bongio, A. (2000). Web Modeling Language (WebML): a modeling language for designing Web sites. *Computer Networks*, 33 (1) 137-157.

[4] Chawla, S., Srivastava, S. (2012, October). A Goal based methodology for Web specific Requirements Engineering. In *Information and Communication Technologies* (WICT), 2012 *World Congress on* (p. 173-178). IEEE.

[5] Chung, L., do Prado Leite, J. C. S. (2009). On non-functional requirements in software engineering. In *Conceptual modeling: Foundations and applications* (p. 363-379). Springer Berlin Heidelberg.

[6] Franch, X. (2006, January). On the quantitative analysis of agent-oriented models. In *Advanced Information Systems Engineering* (p. 495-509). Springer Berlin Heidelberg.

[7] Garrig´os, I (2008). A-OOH: Extending Web Application Design with Dynamic Personalization. Ph.D thesis, University of Alicante, Spain\

[8] Giorgini, P., Mylopoulos, J., Sebastiani, R. (2005). Goal-oriented requirements analysis and reasoning in the tropos methodology.*Engineering Applications of Artificial Intelligence*, 18 (2), 159-171.

[9] Horkoff, J., Yu, E. (2009). Evaluating Goal Achievement in Enterprise Modeling–An Interactive Procedure and Experiences. In *The Practice of Enterprise Modeling* (p. 145-160). Springer Berlin Heidelberg.

[10] Horkoff, J., & Yu, E. (2010). Finding solutions in goal models: an interactive backward reasoning approach. In *Conceptual Modeling–ER* 2010 (p. 59-75). Springer Berlin Heidelberg.

[11] Horkoff, J., & Yu, E. (2011, March). Analyzing goal models: different approaches and how to choose among them. *In*: *Proceedings of the* 2011 *ACM Symposium on Applied Computing* (p. 675-682). ACM.

[12] Horkoff, J., & Yu, E. (2012). Comparison and evaluation of goal-oriented satisfaction analysis techniques. *Requirements Engineering*, 1-24.

[13] ITU-T, I. T. U. T., & Recommendation, Z. (2012). 151 (12/10), User Requirements Notation (URN)–Language definition. *Geneva, Switzerland*..

[14] Jeniffer Horkoff (Accessed on 30/May/2013). jucmNav online Help http://jucmnav.softwareengineering.ca/ucm/bin/view/ProjetSEG/HelpOnLine#Part_IV_GRL_Strategies

[15] Jose Escalona, M., and Gustavo Aragón (2008). NDT. A model-driven approach for web requirements. *Software Engineering, IEEE Transactions on* 34.3 (2008): 377-390.

[16] Kaiya H, Horai H, Saeki M. (2002). AGORA: attributed goal oriented requirements analysis method. In: Proceedings of international requirements engineering conference (RE'02), IEEE Press, p 13.

[17] Koch, Nora, et al. (2008). UML-based web engineering. *Web Engineering: Modelling and Implementing Web Applications*. Springer London, 157-191.

[18] Lai, Alan, Cui Zhang and Senad Busovaca. (2013). 2-SQUARE: *A Web-Based Enhancement of SQUARE Privacy and Security Requirements Engineering. IJSI* 1.1: 41-53. Web. 5 Aug. 2013. doi:10.4018/ijsi.2013010104

[19] Liaskos, S., McIlraith, S. A., Sohrabi, S., Mylopoulos, J. (2010, September). Integrating preferences into goal models for requirements engineering. In *Requirements Engineering Conference (RE), 2010 18th IEEE International*(p. 135-144). IEEE.

[20] Liu, L., & Yu, E. (2001, May). From requirements to architectural design-using goals and scenarios. In *ICSE*-2001 *Workshop: From Software Requirements to Architectures* (*STRAW* 2001) *May* (p. 22-30).

[21] Luo, Hao, and Daniel Amyot. (2011). Towards a Declarative, Constraint-Oriented Semantics with a Generic Evaluation Algorithm for GRL. *iStar*. 2011.

[22] Magableh, B., Barrett, S. (2011). Productivity Evaluation of Self-Adaptive Software Model Driven Architecture. *International Journal of Information Technology and Web Engineering* (IJITWE), 6 (4), 1-19.

[23] Mussbacher, Gunter, and Daniel Amyot. (2009). Goal and scenario modeling, analysis, and transformation with jucmnav. Software Engineering-Companion Volume, 2009. ICSE-Companion 2009. 31st International Conference on. IEEE, 2009.

[24] Paolo Giorgini, John Mylopoulos, Eleonora Nicchiarelli, Roberto Sebastiani(2003) , Formal Reasoning Techniques for Goal Models, S. Spaccapietra et al. (Eds.): Journal on Data Semantics, LNCS 2800, p. 1"20, 2003.

[25] Power, C., Freire, A. P., Petrie, H. (2011). Integrating Accessibility Evaluation into Web Engineering Processes. *Web Engineered Applications for Evolving Organizations: Emerging Knowledge*, 315.

[26] Rahman, W. N. W. A., Meziane, F. (2011). A generic QoS model for web: Services design. *International Journal of Information Technology and Web Engineering* (IJITWE), 6 (3), 15-38.

[27] Roy, JF. (2007). Requirement Engineering with URN: Integrating Goals and Scenarios. M.Sc. thesis, SITE, University of Ottawa, Canada; March.

[28] Salay, R., Chechik, M., Horkoff, J. (2012). Managing requirements uncertainty with partial models. *In: Requirements Engineering Conference (RE),* 2012 20*th IEEE International* (p. 1-10). IEEE.

[29] Chawla, Shailey., Srivastava, Sangeeta. (2012). A Requirements Analysis approach for Web Engineering, *In:* International Conference on Advances in Information Technology and Communication, Dubai, Passent Elkafrawy (Ed.): SPIT 2012, LNICST, Springer, p. 144–149.

[30] Chawla, Shailey., Sangeeta, Srivastava., Malhotra, Deepak . (2013) (In press), Goal based Requirements Analysis using WebURN, *International Journal of Computer Information Systems and Industrial Management.*

[31] Urbieta, M., Escalona, M. J., Luna, E. R., Rossi, G. (2012). Detecting conflicts and inconsistencies in web application requirements. In *Current Trends in Web Engineering* (p. 278-288). Springer Berlin Heidelberg.

[32] Lamsweerde, Van A. (2001). Goal-oriented requirements engineering: A guided tour. *In: Requirements Engineering*, 2001. *Proceedings. Fifth IEEE International Symposium on* (p. 249-262). IEEE.

[33] Lamsweerde, Van A. (2004, September). Goal-oriented requirements enginering: a roundtrip from research to practice [enginering read engineering]. In *Requirements Engineering Conference,* 2004. *In:* Proceedings. 12th IEEE International (p. 4-7). IEEE.

[34] Lamsweerde, Van A. (2009). Reasoning about alternative requirements options. *In:* Conceptual Modeling: Foundations and Applications (p. 380-397). Springer Berlin Heidelberg.