2010

Intelligent computational argumentation for evaluating performance scores in multi-criteria decision making

Rubal Wanchoo

INTELLIGENT COMPUTAITONAL ARGUMENTATION FOR EVALUATING PERFORMANCE SCORES IN MULTI-CRITERIA DECISION MAKING

by

RUBAL WANCHOO

A THESIS

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN COMPUTER SCIENCE

2010

Approved by

Dr. Xiaoqing (Frank) Liu, Advisor
Dr. Maggie Cheng
Dr. Shun Takai

ABSTRACT

Multi Criteria Decision Making (MCDM) is a discipline aimed at assisting multiple stakeholders in contemplating a decision paradigm in an uncertain environment. The decision analysis to be performed involves numerous alternative positions assessed under varied criterion. A performance score is assigned for each alternative in terms of every criterion and it represents satisfaction of the criteria by that alternative. In a collaborative decision making environment, performance scores are either obtained when a consensus can be reached among stakeholders on a particular score or in some cases or controversial when stakeholders do not agree with each other about them. In the previous research an intelligent argumentation system for collaborative decision making was developed. In this thesis; its use is being extended for evaluating performance scores in MCDM. A framework is laid out for using the Intelligent Argumentation approach for resolving controversial performance scores. An application case study of "Selection of a Mine Detection Simulation tool" is used to illustrate the method. To validate it empirically, a case study "to determine division of effort between software quality assurance and software testing," which has a group of 24 stakeholders, is conducted in a hypothetical setup. Its empirical data is collected and analyzed. The analysis serves two basic purposes: 1) to validate capability of the argumentation process in determining the controversial performance scores in MCDM using our intelligent computational argumentation system and to show its effectiveness in capturing rationales of stakeholders and assisting rapid collaborative decision making.

ACKNOWLEDGMENTS

I owe a debt of gratitude to all those who have helped me with this thesis. First I want to thank my advisor Dr. Frank Liu, who supported me in this research. With his professional suggestions and help, I could do this research more efficiently and easily. Secondly, I would also like to express my gratitude to Dr. Shun Takai and Dr. Maggie Cheng for serving on my thesis committee and for taking time to review this work.

This project is supported by the Intelligent System Center (ISC) in the Missouri University of Science and Technology, Rolla.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1. INTRODUCTION

Multi-Criteria decision making techniques and models assist in the process of searching for decisions which best satisfy a magnitude of conflicting objectives. Criterion Satisfaction of criterion by decision alternatives are represented through a performance chart, also called a decision matrix. The decision matrix contains performance scores of different alternatives, which represent their satisfactions for different criteria. The quality of decision is directly related to that of the performance scores. Performance scores are sometimes hard to determine. There is a wide range of different opinions about their values from different stakeholders and they might become controversial in these situations. They require an in-depth analysis of different views and concerns of the stakeholders. As an example, consider a decision problem in which we have three car models from which we need to find the most favorable model in terms of *cost*, *mileage* and *looks*. While it is easy to find performance scores of a car model for *cost* and *mileage*, the quantitative evaluation of car models under the third criteria *looks* is difficult since different stakeholders may have different views about the looks of a car. A consensus may not be easily achieved in this case and a more rigorous approach is needed to find these objective scores.

To deal with the issue of uncertainty in the evaluation of performance scores, many methods make use of linguistic variables in fuzzy logic [9]. Several fuzzy logic based multi-criteria decision making methods are developed using linguistic variables for representing performance scores. They either aid in finding the most favorable alternative from among a set of various alternatives as illustrated in [4] [16] or they determine the alternative with the nearest match to an ideal solution [17]. However, these methods do not resolve the issue; what is the justification behind the performance scores in a decision matrix. Actually, many scores given by stakeholders may not be justifiable. This is why stakeholders need to provide rationales for their performance scores. This arises the need for Intelligent Argumentation.

Intelligent Computational Argumentation is an effective technique for quantifying and measuring subjective factors involved in multi-criteria decision making because it stresses the need for reaching conclusions through logical reasoning. Argumentation based decision making allows stakeholders to provide arguments and justifications as a part of decision process, which in turn increases the speed of agreements being reached. Arguments are intended to support or attack other arguments or decision alternatives. Indeed, an argumentation based approach for collaborative decision making has the advantage of letting a stakeholder specify his views and beliefs along with reasons supporting the same. These reasons may lead their receivers to change their preferences. Consequently, an agreement may be more easily reached with such approaches, when in other approaches (where stakeholder's preferences are fixed) consensus may be more difficult to achieve.

The thesis is aimed at evaluating the derivation of controversial performance scores in an MCDM framework using intelligent computational argumentation. A framework is developed which demonstrates how the intelligent argumentation system can be used to evaluate the performance scores and calculate the favorability of alternatives. A case study is developed to illustrate the working of the developed Intelligent Argumentation technique for resolving controversial performance scores. Also, an empirical study is constituted so as to prove the effectiveness of the intelligent argumentation system in evaluating the performance scores in MCDM.

2. RELATED WORK

For the purpose of this research, two paradigms had to be reviewed. Firstly, the existing techniques for the evaluation of subjective scores in Multi-Criteria decision making had to be studied. Secondly, various argumentation approaches for decision making developed so far had to be reviewed. The following sections provide some insight into the work already done in this regard.

2.1. CURRENT STATE-OF-THE-ART TECHNIQUES FOR EVALUATING PERFORMANCE SCORES IN MCDM

Multi-Criteria decision making paradigm has been supported by many mathematical models in the past. These models evaluate performance scores for alternatives with respect to different criterion. Tsaur, Chang and Yen [16] use the fuzzy MCDM framework for evaluating airline service quality. Their implementation integrates the use of Analytic Hierarchy process (AHP) for evaluating the weights of criteria and fuzzy theory for finding out the performance scores. Wang and Lee [17] develop a fuzzy TOPSIS (Technique for Order preference by Similarity to Ideal Situation) approach based on subjective and Objective weights. They extend the Shannon's entropy method to measure weights in MCDM. The Shannon entropy is a measure of uncertainty in information formulated in terms of probability theory. Chen, Tzeng and Ding [4] also make use of fuzzy MCDM approach to '*Select a service provider'*. They use the pair wise comparison technique for assessing the criteria, and a set of fuzzy linguistic terms for calculating the favorability of the alternatives.

2.2. CURRENT STATE-OF-THE-ART TECHNIQUES FOR AIDING DECISION MAKING USING ARGUMENTS

Philosopher Stephen Toulmin developed a very influential model for argumentation that has guided the development of software tools and systems intended to support the detection and resolution of conflicts in many knowledge domains. Amgoud and Prade [1] extended the argumentation framework to define mathematical models for epistemic and practical arguments. gIBIS (graphical IBIS), represents the design dialog as a graph [5]. While being capable of representing issues, positions, and arguments, gIBIS did not support representation of goals (requirements) and outcomes. IBE [8] extended gIBIS by integrating a document editor. HERMES [14] is a system that aids decision makers to reach a decision by structuring arguments and evidences together in a hierarchy. The evidences are facts which act as a ground for belief and which tend to prove or disapprove the arguments or other evidences. The system assigns weights to the arguments and then evaluates those weights to find the closest alternative to an ideal solution. HERMES is a collaborative system which allows a real time decision making environment for participation among multiple stakeholders.

3. INTELLIGENT COMPUTATIONAL ARGUMENTATION SYSTEM

In the previous research an intelligent computational argumentation system has been developed that allows stakeholders to determine the concerning issue, enumerate the available alternatives, and specify the arguments for those alternatives. The issue signifies a point, matter, or dispute, the decision of which is of special importance to the stakeholders. Alternative positions represent possible choices for dealing with the issue. An argument symbolizes a statement, reason, or fact for or against another argument or an alternative. The decision to be made can also be referred to as '*Strategic Decision*'.

3.1. BACKGROUND

As stated earlier; an intelligent collaborative engineering design system based on argumentation [11] was developed. The design environment supports client-server architecture. On the client side, the system provides user interfaces for solid modeling, annotation, whiteboards for design alternatives, argumentation based conflict resolution, and a chat feature for real time information exchange. On the server side, it manages client communication, concurrent access to design objects, and argumentation network.

The argumentation structure is organized as a weighted directed graph also called a dialog graph [5], as shown in Figure 3.1. The node denoted by circle is a *Position* or an *Alternative* and the nodes denoted by rectangles are *Arguments*. Arrows depict relationship either between two argument nodes or between an argument node and position. The relationship can be either an *Attack* or a *Support*. The strength of an argument is realized by the weight assigned to it. This weight symbolizes the degree of attack or support to an argument or position. The weight value is real number between -1 and 1. A positive weight value implies *support* and a negative weight value expresses *attack*. A weight of zero exhibits *indecision*. For the purpose of implementing the methodology in our current system, we let the stakeholders decide the score of an argument by discussions and reaching a general agreement.

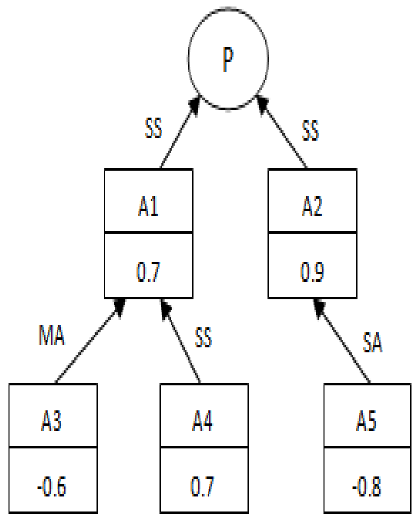

Figure 3.1. Position Dialog Graph

Strengths of arguments are viewed as a fuzzy set and are represented using a set of five linguistic labels. Linguistic labels which are used are *Strong Support*, *Medium Support*, *Indecisive*, *Medium Attack* and *Strong Attack*.

A fuzzy associative matrix is developed and used alongside a fuzzy inference engine, which deal with reducing arguments to a single level and incorporating priority based reassessment for weights of an argument. The two approaches have been discussed in the following sections.

3.2. ARGUMENTATION REDUCTION USING FUZZY INFERENCE ENGINE

The structure of the argumentation tree may become too large and complex to be easily understood and dealt with. A fuzzy inference engine [10] [11] is therefore developed to deal with this problem. It comprises of a set of 25 fuzzy rules which deal with the task of reducing the arguments to a single level. These rules assess the impacts of indirect arguments on alternative positions using fuzzy logic. The argument reduction follows a set of four general heuristic rules [10] [11]:

Argument Reduction Rule 1: If argument B supports argument A and argument A supports position P, then argument B supports position P.

Argument Reduction Rule 2: If argument B attacks argument A and argument A supports position P, then argument B attacks position P.

Argument Reduction Rule 3: If argument B supports argument A and argument A attacks position P, then argument B attacks position P.

Argument Reduction Rule 4: If argument B attacks argument A and argument A attacks position P, then argument B supports position P.

Based on the above set of heuristic rules, twenty-five fuzzy argumentation inference rules are generated. These rules are specified in a matrix called as *fuzzy association memory matrix* (FAM); which takes into account the five linguistic variables already defined. The fuzzy inference engine uses the fuzzy association matrix and takes two inputs to produce one output. One of the inputs is the strength of argument to be reduced and the other input is the strength of argument right above it. The output is the reduced strength of the argument. Details of the Fuzzy Association Matrix are provided in [10] [11] and are constituted as a part of the previously done research.

Figure 3.2 shows the argumentation tree in its initial form as put in by the stakeholders and its reduced form after applying the fuzzy association rules [10] [11]. The labels with prefixes 'O' in the figure represent different stakeholders. The detailed process for argumentation reduction via the fuzzy inference engine is explained in [11].

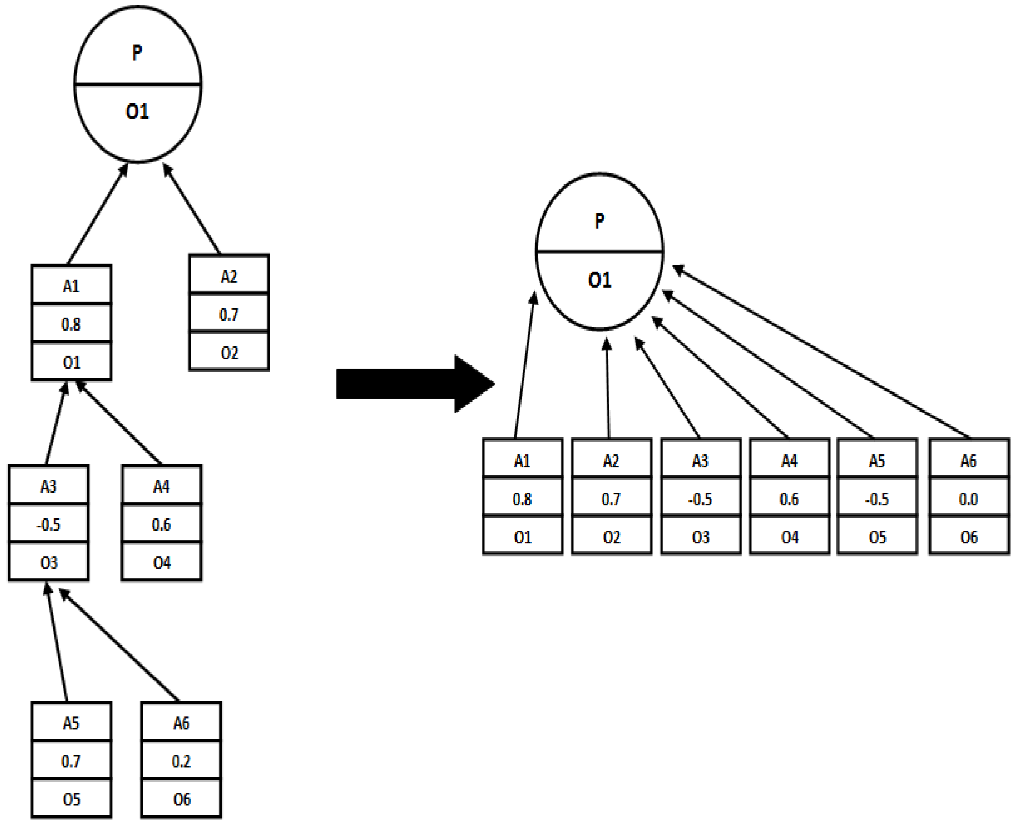

Figure 3.2. Argumentation Reduction

3.3. INCLUSION OF PRIORITY OF STAKEHOLDER IN INTELLIGENT ARGUMENTATION

Priority of a stakeholder represents an authoritative rating that establishes precedence of a stakeholder over other stakeholders. Every stakeholder is assigned a priority [10] which is nothing but a value between 0 and 1. The higher the value of the priority for the stakeholder, the higher is his importance in the argumentation system. Priority is incorporated into the decision making either through weighted summation method or through priority based reassessment of argument's strength using fuzzy logic [10]. Priority of a stakeholder is decided by the stakeholders, in the same way as the weights of arguments are decided.

3.3.1. Weighted Summation. After reducing the argumentation tree to a single level using fuzzy association matrix as explained in [11], the favorability factor of an alternative are calculated using the weighted sum of strengths of arguments and priority as follows:

$$\text{Favorability} = \sum_{i=1}^{m} p_i \times w_i$$

In the equation above 'w_i' is the strength of an argument at the highest level achieved after reduction and 'p_i' is the priority of the stakeholder who raises the argument.

3.3.2. Reassessment of Argument's Strength Based On Stakeholder's Priority. Reassessment of an argument's weight using priority of a stakeholder follows a set of priority reassessment heuristic rules [10]:

General Priority Re-assessment Heuristic Rule 1: If the stakeholder specifying an argument A has a higher priority, the strength of the argument should be higher than it is.

General Priority Re-assessment Heuristic Rule 2: If the stakeholder specifying an argument A has a lower priority, the strength of the argument should be lower than it is.

A set of 3 linguistic variables are used to represent the priorities of a stakeholder. The linguistic variables used are high (H), medium (M), low (L) [10]. These 3 linguistic variables are combined with 5 linguistic variables stated in figure 2 to produce a set of 15 new heuristic rules. The detailed process for reassessment of arguments weight using priority is given in [10].

4. FRAMEWORK FOR ASSESSING THE PERFORMANCE SCORES IN MULTI CRITERIA DECISION MAKING THROUGH INTELLIGENT ARGUMENTATION

Argumentation can be exclusively used for evaluating those performance scores in a multi criteria decision framework for which the objective interpretation is debatable among various stakeholders. These scores are controversial in nature and require deep analysis and rigorous discussions before they can be developed into a quantifiable value. Argumentation between stakeholders provides a logical way to figure out these subjective performance scores and the scores thus obtained represent a consensus.

4.1. ELEMENTS OF THE ARGUMENTATION SYSTEM

The argumentation system consists of attributes which aid in decision making in a multi criteria decision making domain. The element '*Criterion*' is a new addition as a part of this thesis .The list of all the elements of the argumentation system is presented in the Table 4.1 below.

Table 4.1. Elements of the Intelligent Argumentation System

Name	Description
Stakeholder	People who establish an issue to be dealt with
Issue	A decision problem for which various considerations are laid
Criterion	A rule or principle for testing or evaluating an alternative
Alternative	A given possible approach for resolving the Issue
Argument	Views and opinions of different stakeholders targeting a specific alternative-criterion pair
Evidence	A fact that lays stress on the argument at hand
Weight of an Argument	Degree of attack or support between -1 and 1
Priority of a Stakeholder	Authoritative rating that establishes precedence of a stakeholder over another. Its value is between 0 and 1

4.2. DATABASE

In the developed argumentation system, the database is used to permanently store the data pertaining to various projects. As a part of previous research, the elements such as '*Project*', '*Issue*', '*Position*', '*Argument*', and '*Evidence*' were there in the database. In the current research, some of the previous elements were changed and a new element '*Criterion*' was added. Figure 4.1 shows the database schema after the incorporation of criteria was done into the application.

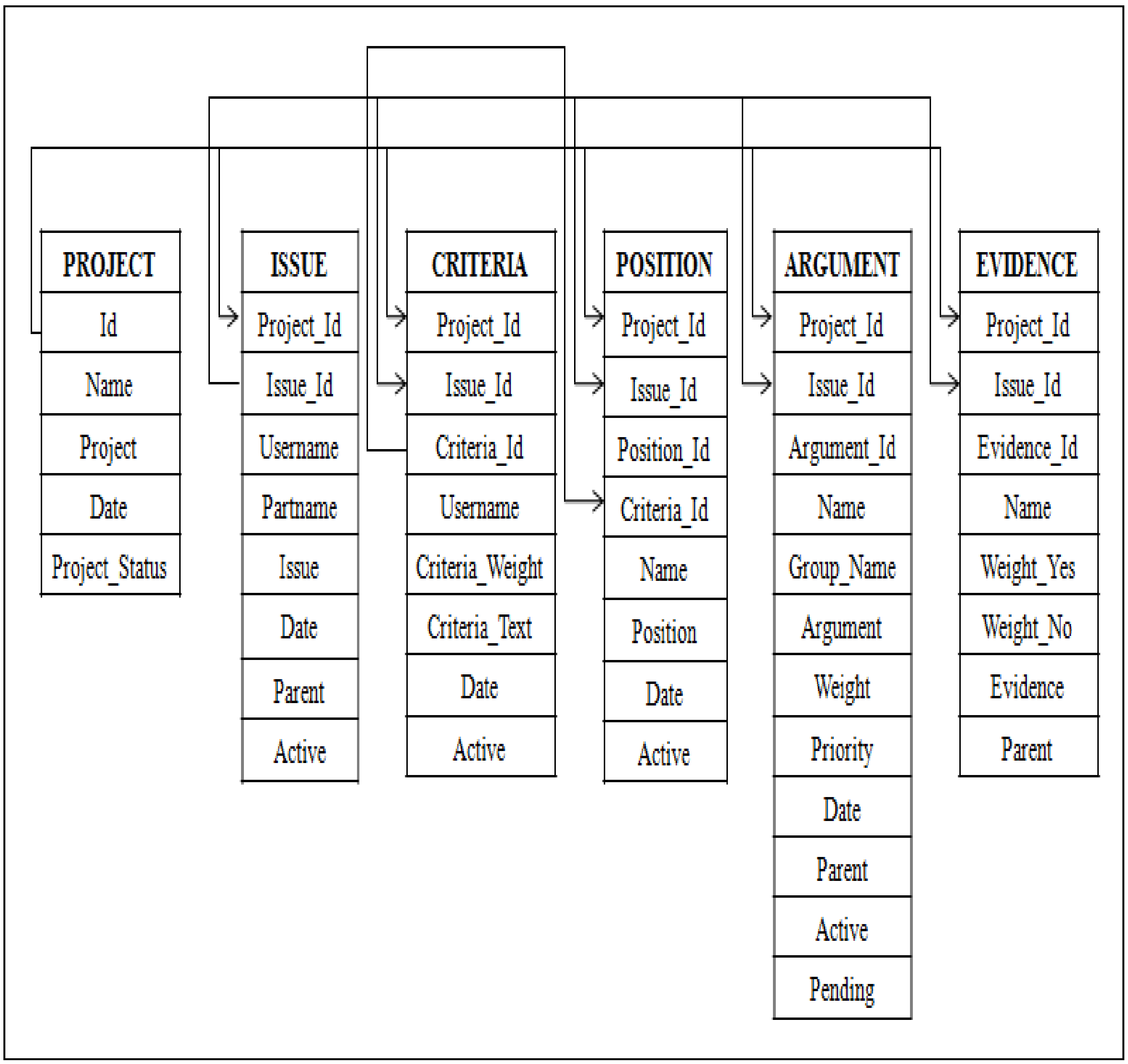

Figure 4.1. Database Schema after Incorporation of Criteria

The descriptions of the various attributes of the '*Criteria*' tables are illustrated in table 4.2:

Table 4.2. Attributes of the Criteria Table

Attribute Name	**Description**
Project_Id	Associated Project ID
Issue_Id	Associated Issue ID
Criteria_Id	Criteria ID
Username	Name of the stakeholder entering the criteria details
Criteria_Weight	The weight of the criteria
Criteria_Text	The text associated with the criteria
Date	The date when the criteria is entered in the database
Active	Status of criteria(i.e. whether it is open or closed)

4.3. ARGUMENTATION PROCESS

The argumentation framework for multi-criteria decision making involves multiple stakeholders who establish an issue to be dealt with. Issue here symbolizes a point, the decision of which determines a matter. The issue serves as a decision problem for which various considerations are laid. The stakeholders decide the criteria set upon which the alternatives are to be analyzed. The alternative positions are evaluated against the entire criteria set and their respective performance scores are given in the decision matrix. The argumentation framework for evaluating performance scores is illustrated in figure 4.2. The rectangle boxes describe the objects involved in the decision making framework whereas the tilted rectangle boxes show the input and output associated with the system. The diamond shaped box shows the possible branching condition and the arrows exhibit the possible transition between various objects. As can be deduced from the figure, we can make use of the argumentation system whenever the stakeholders cannot reach a consensus in deciding the performance scores.

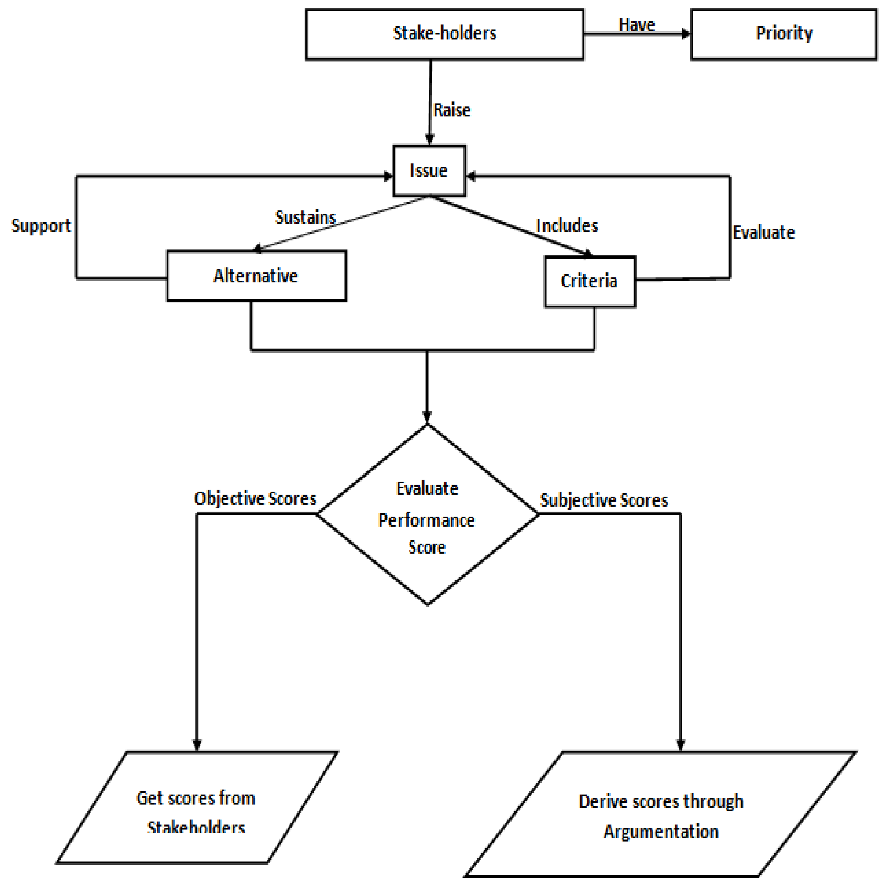

Figure 4.2. The Argumentation Process

Figure 4.2 shows the criteria specific argumentation that takes place once the need for deriving performance scores through argumentation arises. Every alternative is scored against every criterion in the decision matrix. The arguments are entered in to the system by the stakeholders target a specific criterion for an alternative. The argumentation input itself consists of four inputs i.e. the name of the stake holder, the argument specification, the weight of the argument and the priority of the stakeholder. The name of the stake holder specifies the identity of the person specifying the argument. The argument is a sentence in English language which portrays the views and opinions of the stakeholder. The weight of the argument represents the degree of attack or support of that argument for that criterion.

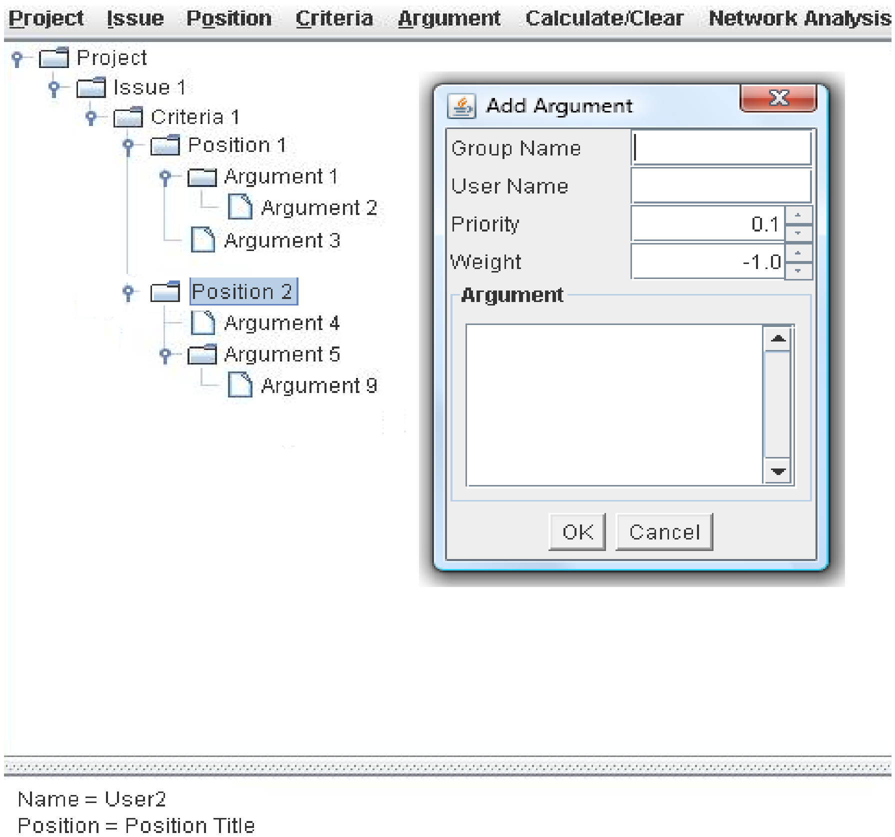

Figure 4.3. The Intelligent Argumentation Process in our System

Figure 4.3 demonstrates how the addition of an argument is handled in our present system. The input for '*Group Name*' and '*User Name*' correspond to a specific user and his group. The rest of the entries i.e. priority and weight are the same as discussed previously. The terms '*Alternative*' and '*Position*' are synonymous in the system.

4.4. MULTI CRITERIA ARGUMENTATION TREE STURCTURE

Table 4.3 shows a sample decision matrix in which various alternatives are evaluated against different criterion. The scores specified in the decision matrix can be either '*Easily Determined*' or '*Controversial*'. Figure 4.4 shows argumentation tree that will be developed for evaluation of subjective scores for the decision matrix in Table 4.3.

Table 4.3. Sample Decision Matrix

Performance Chart

	Criteria1	Criteria2	Criteria3
Alt1	Easily Determined	Controversial	Controversial
Alt2	Easily Determined	Controversial	Controversial
Alt3	Controversial	Controversial	Easily Determined

Every criterion in Table 4.3 has an alternative as a child node. A detailed argumentation tree is developed for this alternative. In this way, the arguments are stated targeting this very criterion. This pin-points the arguments being stated to the highlighted matter and can extract the finer details of the concerning issue.

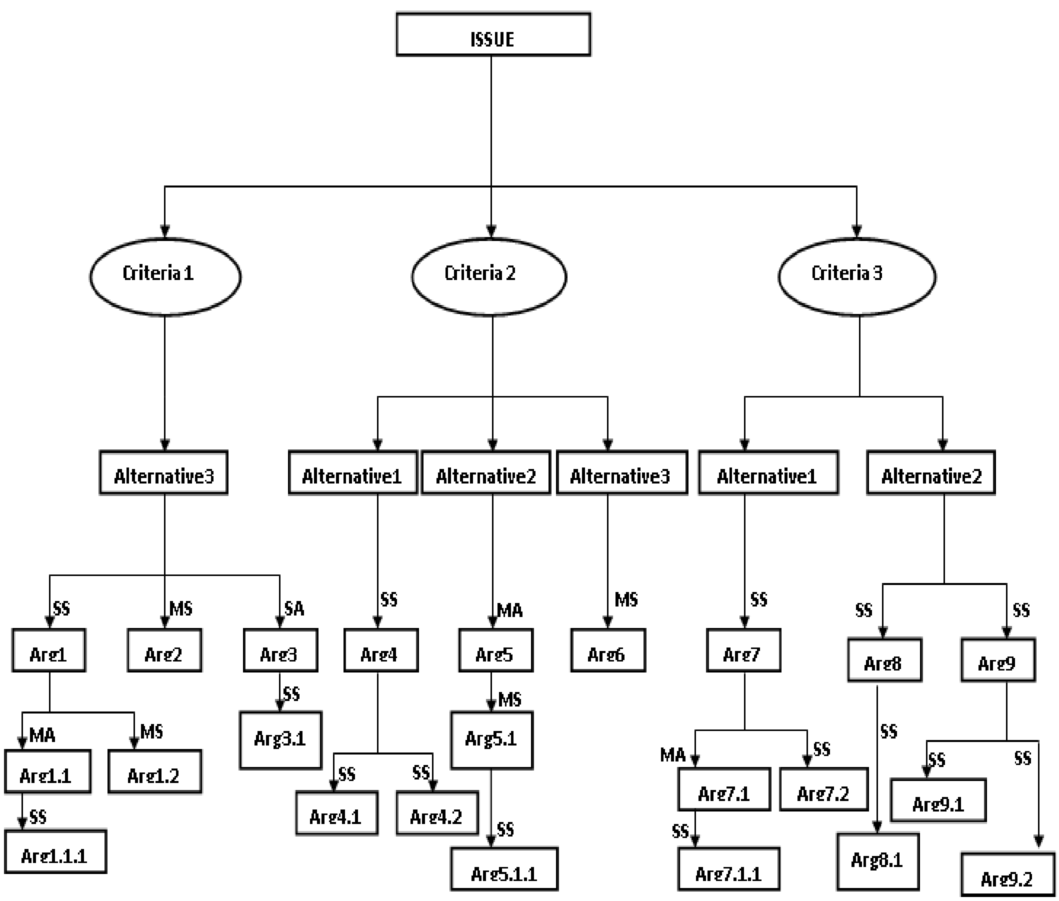

Figure 4.4. Sample Argumentation Tree

Using the above sample tree approach, we can evaluate the performance score of every alternative using the techniques of fuzzy association matrix to derive the impact of indirect arguments and aggregating the scores derived from the argumentation sub-tree with the weight of the criteria to derive the performance scores in the decision matrix.

5. CASE STUDY: SELECTION OF A SOFTWARE PLATFORM FOR THE DEVELOPMENT OF MINE DETECTION SIMULATION

As a part of our investigation, a project was selected which was being implemented at the Mechanical and Aerospace Engineering Department of Missouri University of Science and Technology, Rolla. The project served as an ideal platform for us to test our proposed technique because the issue involved a decision problem which was under review of concerned research team. The issue had certain evaluation criteria which were laid down by the team members themselves. Every criterion in the criterion set had a favorability factor or weight which represented the role that criteria played in deciding the winning alternative. We decided to use Analytical Hierarchy Process (AHP) for prioritizing the criteria due to its wide use and acceptability. The alternatives to be evaluated were also proposed by the team members or more so; the stakeholders. The nature of the issue involved selecting the most favorable alternative from the set of underlined alternatives. The alternative positions were first evaluated on the specified criteria. Each alternative was given a quantified value which indicated its degree of satisfaction for the given criterion. This score was given by stakeholders after rigorous discussions and reviews. The values were then put in the decision matrix and the favorability of all the alternatives was calculated using the sum of products of the weights of criteria and the score of an alternative with respect to that criteria. The alternative with the highest favorability score was selected. The following sections describe the steps taken to resolve the issue for this research project.

5.1. DESCRIPTION

The artifacts presented at the time of decision making were; the issue, the criterion set, a set of alternatives and concerned stakeholders. The subsections describe how and why the artifacts were important for decision making.

The Issue - The issue deals with selecting a suitable software platform for developing a *Mine Detection Training Software*. The development of mine detection training tool is a research project supported by the US army. The project deals with simulating the real world conditions while detecting mines; into a system which could be used for training purposes. The development of this kind of a simulated tool has a lot of advantage over the previously developed systems. The systems already present for mine detection and training purposes are very basic and do not have much flexibility in terms of customization. The simulated environment for such an environment will require an efficient software tool which can display the performance of a trainee on a real time basis. The system will also support some very important functionalities like recognizing different intensities of sound generation on identification of different kinds of mines, a control panel for customizing the simulated mine field and generation of reports showing quantitative summary performance in terms of coverage rate statistics, covered area statistics and mine target location.

The Stakeholders - The stakeholders involved in the decision process are two students from the '*Mechanical and Aerospace Engineering department*' who are responsible for developing the sound detection algorithm. Another stakeholder is a '*Computer Science*' student responsible for developing the software tool. This software tool will help the trainer evaluate a trainee's performance with respect to functionalities defined above. Two professors, one from Computer Science and another from Mechanical Engineering department act as stakeholders responsible for overlooking the overall software development process and playing a crucial role in making important decisions.

The Criterion Set - As stated earlier, criterion is the attribute for which favorability of an alternative is calculated. The criterion set defined for this case study comprises of *Reusability*, *Meeting Operational Requirements*, and *Meeting Project Deadline*. By reusability, we mean, the amount of reuse of different functionalities that can be achieved from the previously developed system on Mine detection training tool. Meeting operational requirements implies how effectively a desired operational capability can be satisfied by an alternative. For example, some alternative might lack a certain operational capability like database support whereas another may support it with enhanced features. Meeting project deadline stresses on the fact, whether the project requirements can be satisfactorily achieved within the stipulated deadline which in our case was around one year.

The Alternatives - To resolve the concerned issue, the stakeholders decided to choose one software platform for developing the mine detection training tool among the three stated alternatives. *Adobe Director*, *Adobe Flash*, *Open GL* were chosen as the three possible alternatives along with some justifications. Adobe Flash was chosen as one of the alternatives because; the stakeholders already had a previous developed system for mine detection developed using Adobe Flash. One of the considerations involved here was to enhance this system rather than develop a new system from scratch. Same reason applied to choosing Adobe Director as one of the other alternatives. Open GL was picked up as one of the three alternatives in the case when a new development had to be started. Open GL is an advanced software development platform and it could have served as a good platform for the mine detection training system. Figure 5.1 summarizes the project and its three alternatives positions available.

Figure 5.1. Mine Detection System Along With Three Alternatives

5.2. PRIORITIZING THE CRITERIA

For an effective decision making, we had to weigh the criteria according to their importance in the decision making process. For this, we choose Analytic Hierarchy process because of its effectiveness in performing pair wise comparison of elements .Table 5.1 shows the ranking table used for comparing the two criteria.

Table 5.1. Criteria Comparison Table For AHP

Value a_{ij}	**Comparison Description**
1	**Criteria i and j are of equal importance**
3	**Criteria i is weakly more important than j**
5	**Criteria i is strongly more important than j**
7	**Criteria i is very strongly more important than j**
9	**Criteria i is absolutely more important than j**

Table 5.2. Comparison Values for Prioritizing Different Criteria

	Reusability	**Meeting Operational Requirements**	**Meeting project Deadline**
Reusability	**1**	**1/5**	**3**
Meeting Operational Requirements	**5**	**1**	**7**
Meeting Project Deadline	**1/3**	**1/7**	**1**

Table 5.3. Normalized Criteria Comparison

	Reusability	Meeting Operational Requirements	Meeting project Deadline
Reusability	**0.158**	**0.148**	**0.272**
Meeting Operational Requirements	**0.789**	**0.746**	**0.636**
Meeting Project Deadline	**0.053**	**0.106**	**0.092**

Table 5.2 and Table 5.3 show the weight values of the three criterions as compared to each other using the AHP process. These weights have been decided by the stakeholders after discussions among themselves. Average weights can be derived from Table 4 as follows:

Reusability- 0.193,

Meeting Operational Requirements- 0.724

Meeting Project Deadline- 0.083

These weights represent the priority of each criterion on a scale of 0 to 1.

5.3. ARGUMENTATION TREES

Argumentation trees are developed for each and every alternative separately. The arguments are stated by stake holders and assembled under the alternative but they target a specific criterion. These arguments can either be supporting or attacking each other or their respective alternative nodes. We present three figures, where each figure represents the argumentation hierarchy for one alternative. Rectangular boxes represent the alternatives with the name of the alternative under it. Ovals represent the criteria with their description. The arguments are specified by labels ‘A’, ‘B’, ‘C’ for alternative “Adobe flash”, “Adobe Director” and “Open GL” respectively. Along with the labels, the arguments also have indexes associated with them. Beneath the labels are two boxes. The box on left shows the weight of the argument whereas the box on right shows the priority of the stakeholder who specifies the argument. Once the argument has been specified, the user enters its weight. We first reassess the weights of the arguments using priority reassessment discussed in [10].Then using the techniques specified in [11], we reduce the arguments to a single level. Finally, the weighted summation of the arguments with the criteria weights helps us evaluate the final weights for the decision matrix. It is important to note here that, the aggregation method used for calculating the favorability is a weighted summation. The three argumentation hierarchies for the three alternatives are presented in the Figures 5.2, 5.3, and 5.4. The diagrams contain arguments, their weights and the stakeholder's priorities.

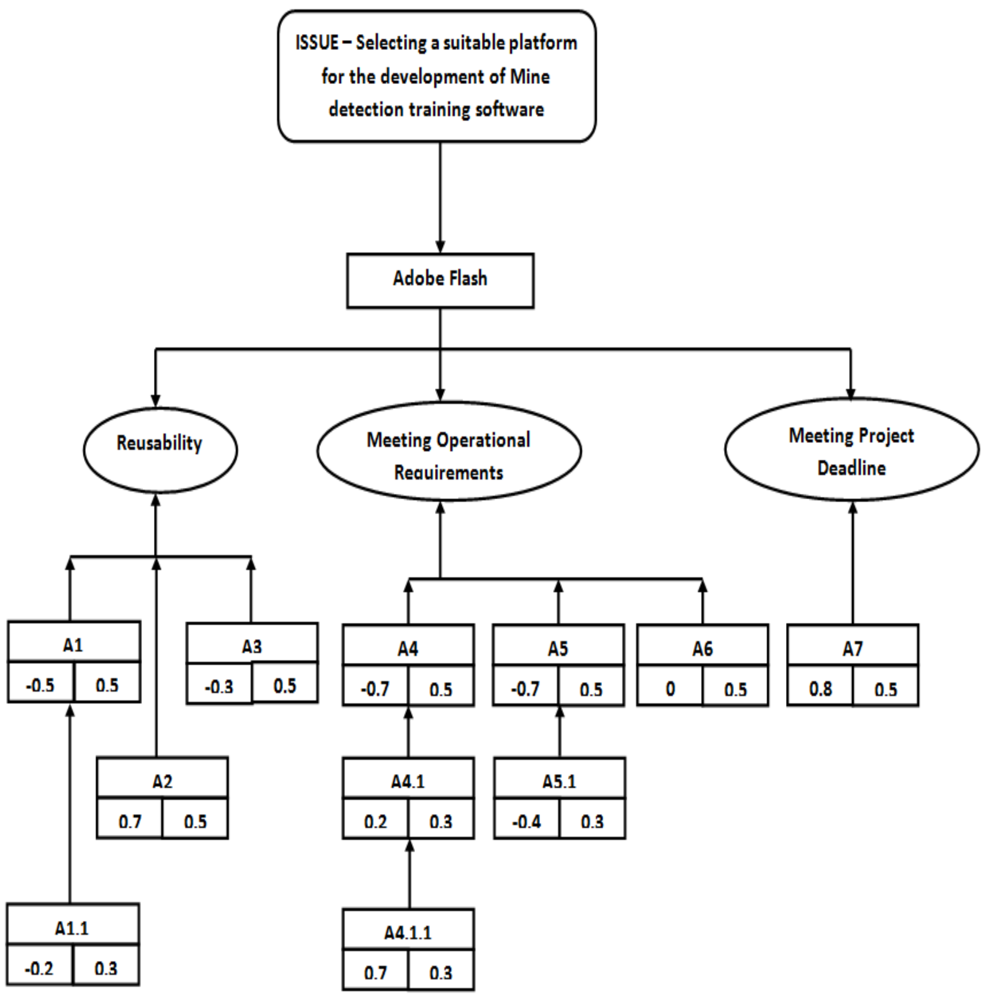

Figure 5.2. Argumentation Tree For Adobe Flash

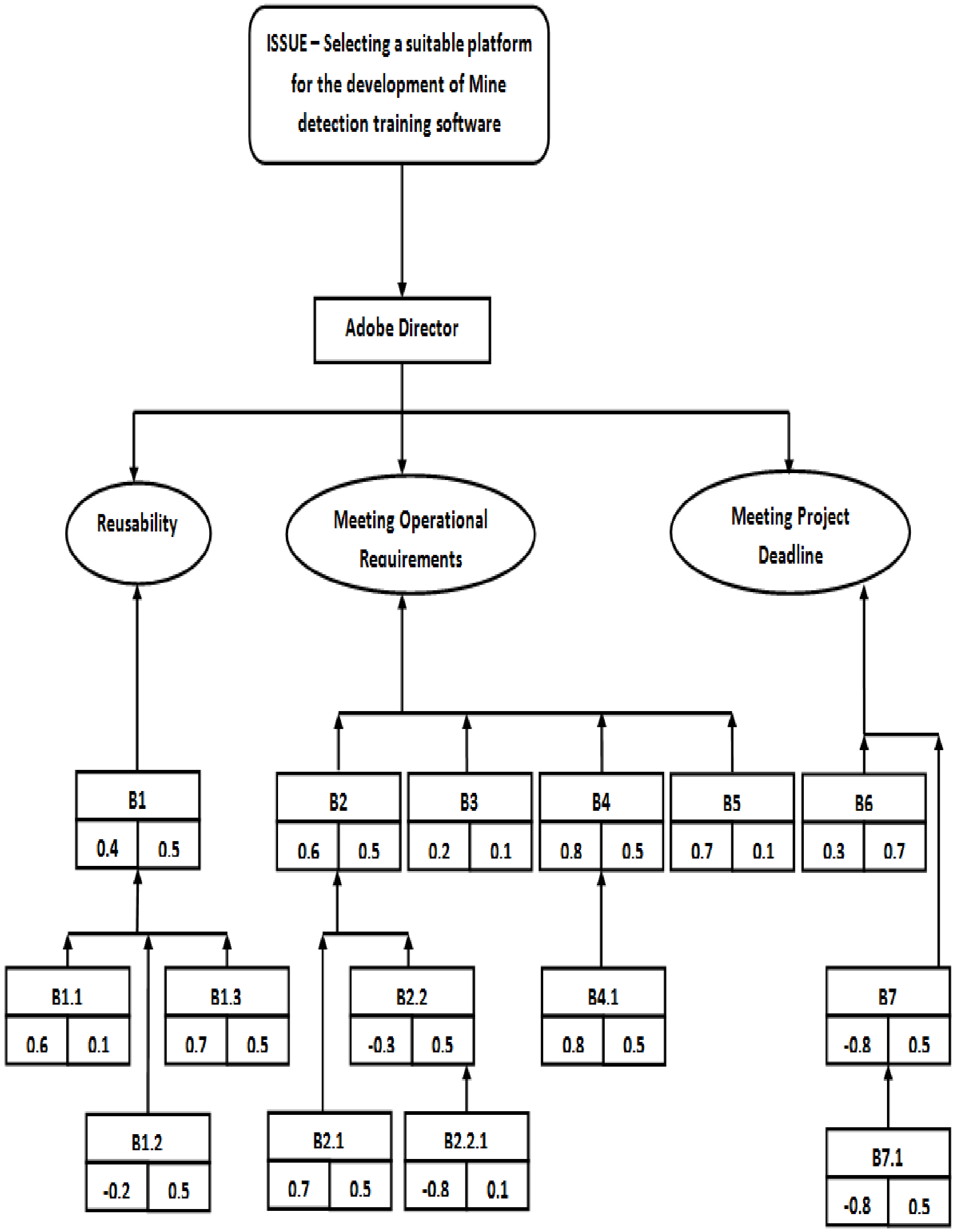

Figure 5.3. Argumentation Tree For Adobe Director

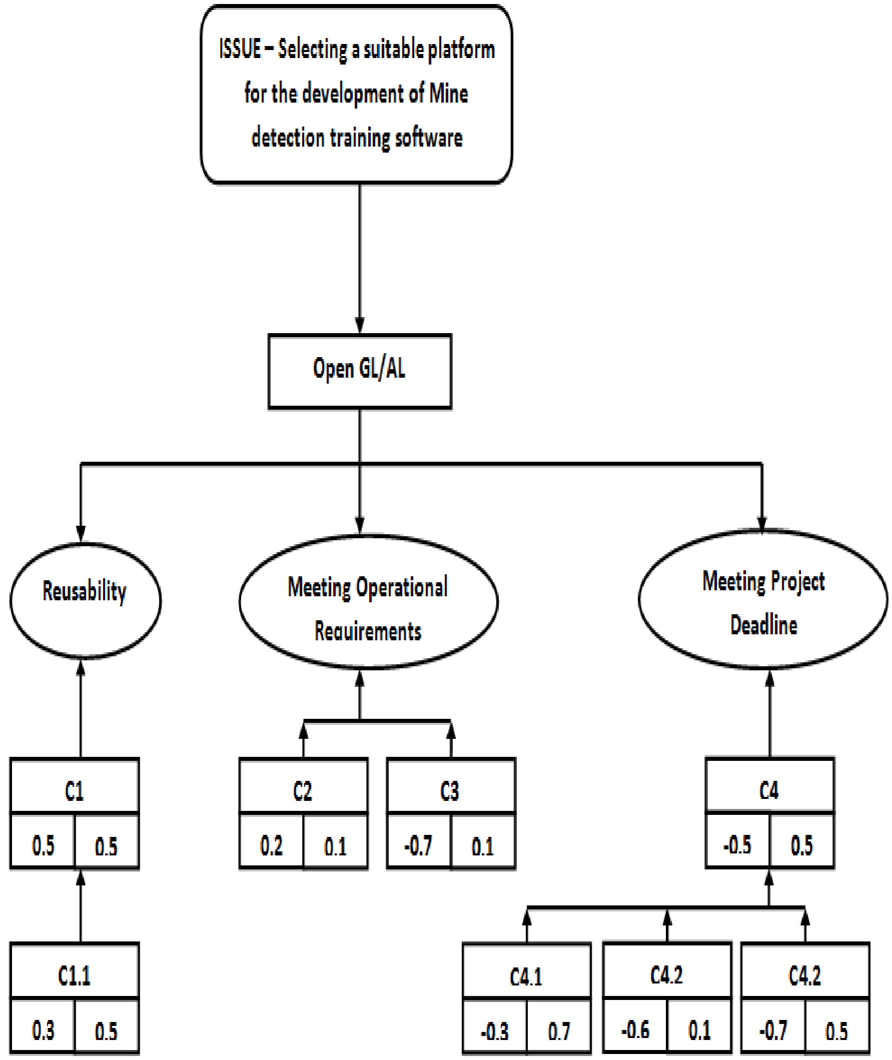

Figure 5.4. Argumentation Tree For Open GL

A1-The current system in flash does not have the functionality of dynamic allocation of particles like mine or clutter. It places them randomly**.**

A1.1-That is not of much importance because it still gives a new position to mine and clutter particles.

A2-Current system in flash has faster response time as compared to system in Adobe Director.

A3-The current system doesn't satisfy many of the features required for the new system like database.

A4-Adobe Flash cannot communicate with database.

A4.1-Flash doesn't support database but database support is very important and critical.

A4.1.1-The system should be able to generate evaluation reports for trainee based on previous records stored in the database.

A5-Flash doesn't create sound clips.

A5.1-We don't need sound creating features as the system has to generate sound. We can play externally recorded sound files using Adobe Flash

A6-Flash can provide good visual effects as compared to Adobe Director

A7-The developer has good knowledge in development using Flash so the system can be developed quickly

B1-We could reuse the system already developed for sound generation, as it is developed using Adobe Audition for analysis which is somehow related to Adobe Director

B1.1-The current system is better synthesized in terms of sound production and the sound produced is also instantaneous rather than discrete

B1.2- That current system has certain performance issues like slow response time

B1.3- The current system in Adobe Director has the feature of producing dynamic coloring scheme on approaching a mine. This kind of scheme is highly preferable and is not present in Adobe Flash system

B2- Adobe Director can provide more functionality as compared to the current flash system. E.g. Multiple sounds while detecting mines**.**

B2.1-Adobe Director can provide better visual effects as compared to flash e.g. in case of GUI's**.**

B2.2- A modified version of the current system in flash can also provide the same functionality

B2.2.1- We cannot integrate code developed in other platforms with Flash, but Flash can be integrated in Adobe Director

B3-The interface provided by flash is not professional enough. It is too simple and straight forward for doing more things in future**.**

B4- Easily available plug-ins can help integrate the tracking system developed in C# with Adobe Director.

B4.1-Code developed in Open GL/AL can also be integrated using Adobe Director using suitable stubs.

B5- A new sound recognition algorithm is being developed in Adobe Audition which can be integrated with Adobe Director but not with Open GL or Flash (Evidence supported)

B6-If the current system is reused; the project deadline can be met easily.

B7-The developer has very little experience in development using Adobe Director**.**

B7.1-The developer can take help from the already developed system in Adobe Director.

C1-The tracking software already developed is coded in C#/NX5. We could reuse that and develop our system in Open GL/AL

C1.1-Open GL has C# libraries which can be used to develop the system

C2-Because the platform used is for high end application development, it can provide good GUI and database support

C2.1-Open GL/AL can help us generate dynamic surfaces for mine detection and training which the original system in flash does not have.

C4-Open GL does not support connectivity with Adobe Audition. Adobe Audition is required for creating sound recognition algorithm

C3-Open GL does not support connectivity with Adobe Audition. Adobe Audition is required for creating sound recognition algorithm

C4-The time taken for developing the project using open GL will be comparatively more as the whole system would have to be developed from scratch.

C4.1-If Open GL has support for C# libraries, and then the system could be developed faster as developer is quite familiar with programming languages like C#.

C4.2-Open GL has excellent documentation that could help the developer learn the platform with ease.

C4.3-Developer has very little experience in working with Open GL platform.

5.4. RESULT

For the case study, alternative B i.e. Adobe Director was the most favorable alternative amongst all the three. It catered to the *Meeting Operational Requirements* criteria quite well and aimed at meeting most of the desired operational requirements for the system. Its calculated value was higher than the other two alternatives. Table 5.4 shows the performance scores derived for various alternative- criteria pairs for this case study.

Table 5.4. Performance Scores Derived Through Intelligent Argumentation

	Reusability (0.193)	**Meeting Operational Requirements (0.724)**	**Meeting Project Deadline (0.083)**
Adobe Flash	**-0.02**	**-0.621**	**0.4**
Adobe Director	**0.46**	**1.205**	**0.13**
Open GL	**0.5**	**-0.05**	**0.4**

Favorability of an alternative calculated using weighted summation:

$$EvaluationScore(A_j) = \sum_{i=1}^{n} Wc_i * P(A_j, C_i)$$

W_{A_j} = Evaluation Score of Alternative j

W_{C_i} = Weight of an Criteria i

P (A_j, C_i) = Performance Score for Alternative 'j' and Criteria 'i'

n = 3 (Number of Criterion)

Evaluation Score (Adobe Flash) = -0.420264

Evaluation Score (Adobe Director) = 0.97199

Evaluation Score (Open GL) = 0.0935

6. EMPIRICAL STUDY:TO DETERMINE DIVISION OF PERCENTAGE OF EFFORT TO BE APPLIED BETWEEN SOFTWARE TESTING AND SOFTWARE QUALITY ASSURANCE FOR A HYPOTHETICAL SCENARIO

6.1. OBJECTIVE

The empirical study has two major objectives:

i) The first objective is to use intelligent argumentation system for assessing controversial performance scores in a multi-criteria decision making.

ii) The second objective is to validate the overall effectiveness of the intelligent argumentation in capturing rationale of stakeholders.

It has 25 participants. Its results are analyzed empirically.

6.2. THE APPROACH

In the beginning, a group of 25 students of a software testing and quality assurance class were briefed about the issue: “Determine the %division of effort between Software Quality Assurance and Software Testing” in a large organization. Its background and the MCDM elements, such as criteria and decision alternatives were discussed. The experiment was conducted in three phases.

In the first phase, the students were required to go through background documents about the case study and complete a survey. The purpose of the survey was to capture their initial thoughts regarding the issue and their preferences of solution alternatives. The survey had a set of two questions which basically asked for their choice of selection and the reason for their choice. The time period for the first survey was one week.

After the data was collected for the first survey, the students were given the access to the Argumentation system. The argumentation system is a client server base system which his developed using JAVA and is supported by a MySql database. The argumentation system was run using one of the servers in the software engineering laboratory in Missouri S&T, Rolla. Each student was given a username and a password for working with the argumentation system. The password was an identification number which was unique to every student. This helped in identifying the thought process of each and every student and also helped in separating his arguments from the rest of the group for analysis. The details of the argumentation process and the analysis done on the same are mentioned in the following sections. A total of 218 arguments were received in a span of three weeks. After the completion of the Argumentation process, our system calculated the favorability of the alternatives. The argumentation system was monitored continuously by the mediator. A mediator was a person who was assigned the task of checking the arguments status and deal with various issues such as correctly placing misplaced arguments, handling student queries in operating the argumentation system, and correcting incorrect weights. Weights for those arguments were changed which were either misplaced or were too high to too low based on the discussion between the mediator and the stakeholder. Calculation of favorability was followed by a rigorous analysis of the data gathered during the argumentation process. The findings of the analysis are mentioned in the following sections.

After the argumentation process was over, the students were once again asked to complete a second survey. Before the students could actually fill the second survey they were required to go through the arguments in the argumentation system. The survey had four questions. The first two questions were the same as in first survey. The next two questions were there to gather data such as the number of arguments they reviewed and the shift in their opinion since the first survey. The details of the second survey are provided in the following sections. Figure 6.1 presents an overview of the whole process involved in this empirical study.

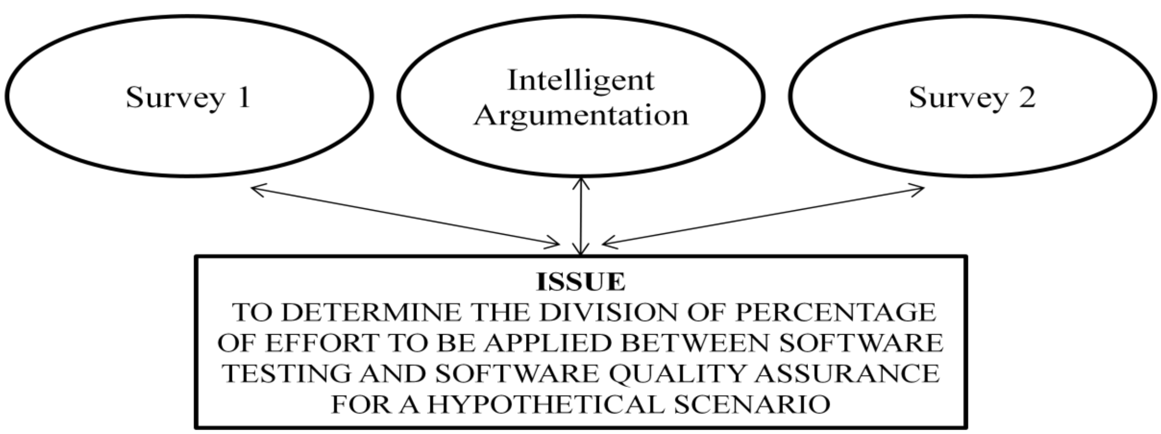

Figure 6.1. The Three Main Steps in the Empirical Study

All the data gathered was analyzed and compared so as to observe the thought process of the students in dealing with the particular issue. The results were used to validate the idea that Intelligent Argumentation could aid in resolving the controversial performance scores in a Multi Criteria Decision Making domain effectively and was a really helpful tool in the decision making process. Figure 6.2 presents detailed steps for each stage in the empirical study process.

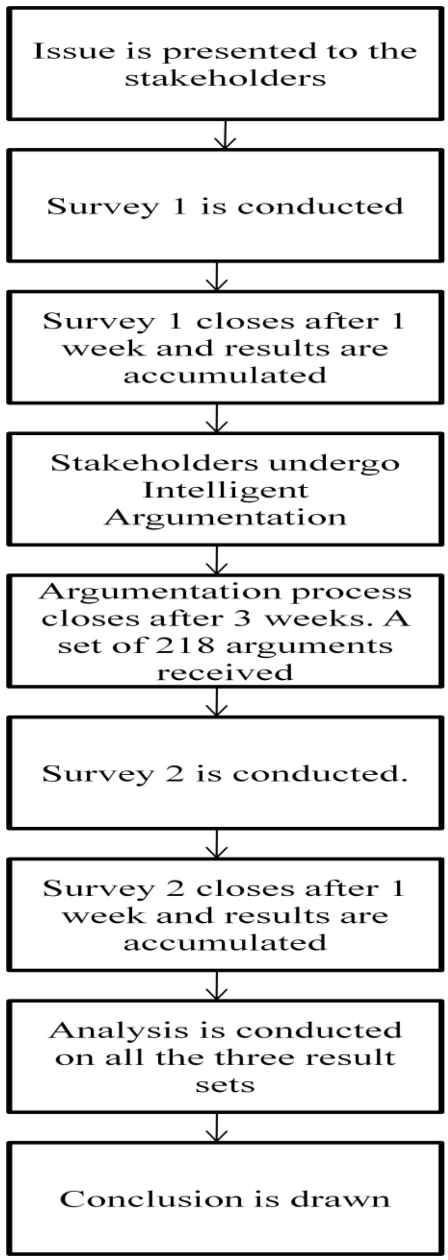

Figure 6.2. Flowchart Showing the Steps in Empirical Study

6.3. BACKGROUND INFORMATION – HYPOTHETICAL SCENARIO

The case study undertaken is a hypothetical scenario which consists of a large scale IT firm and a bank with growing business. The IT firm serves as client for the bank and the bank serves as customer for the IT firm. The whole setup is hypothetical one and the empirical study is designed around it. The background information for both the parties i.e. the bank and the IT firm was given to the students participating in the decision making process. The participants had to resolve the given issue in figure 6.1 based on the background information given to them. The rationale behind giving the background information was to strictly limit the argumentation process to facts in the information.

HSBS Corporation Bank (Hypothetical Customer). HSBS Corporation Bank is one of the growing national banks in United States with a presence in around 30 major cities across US with an annual turnover of 200 billion $. The bank has a customer base of around 200,000 people in United States. The bank has a great reputation in the market and has plans for extending its reach to international locations. HSBS corporation bank has its national headquarters at St Louis, Missouri. The Headquarter is connected to all the local branches in United States and controls the major policy decisions. Hence, all the branch offices are interconnected through the Headquarter at St Louis.

The present structure of the bank is shown in figure 6.3.Every branch office has a database server that is used to store the records local to that branch. The information stored at servers of local branches is employee information, customer information, account information, transaction information, payroll information, and loan information. The information stored at the headquarters is the replica of all the branches as well information pertaining to itself. Whenever a local branch needs to connect to other local branches of different cities, it does so by communicating with the headquarters. The Headquarter can either relay the information from one local branch to other or send information from its own database.

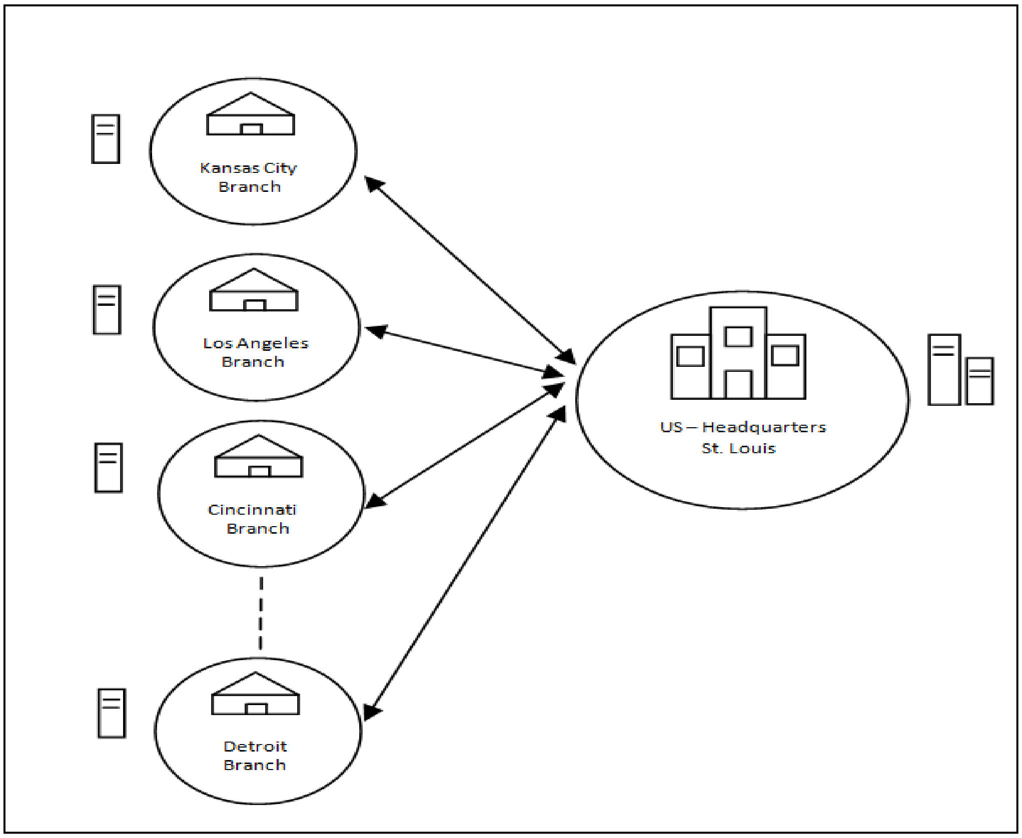

Figure 6.3. Structure of HSBS bank

The software application that is presently used by HSBS bank employees is a mainframe application (like DB2) which does not support interactive graphics and is not very easy to use. The present system connects all the branch offices to each other and allows the employees of different branches to exchange information but does not interact with a customer directly. All the operations are done on a hand to hand basis. So for example, if some account holder in the bank needs to deposit money, he should be physically present at the bank's local branch and do the transactions.

Hence, there arise a number of motivations for moving to a new system. Due to rising business profits, the bank plans to open more branches elsewhere in the country. The bank wants to expand their presence in Europe, with a Headquarter in London. They also want to connect the Headquarters both in US and UK. Along with that they develop e-banking facilities for its customers. They also want to move to a newer database technology (like Oracle…etc.) that could be compatible with the e-banking web application.

As stated above, the managing committee lays out a plan for company expansion and contacts HOBNOB Inc, a software company to develop a system for it as shown in figure 6.4.

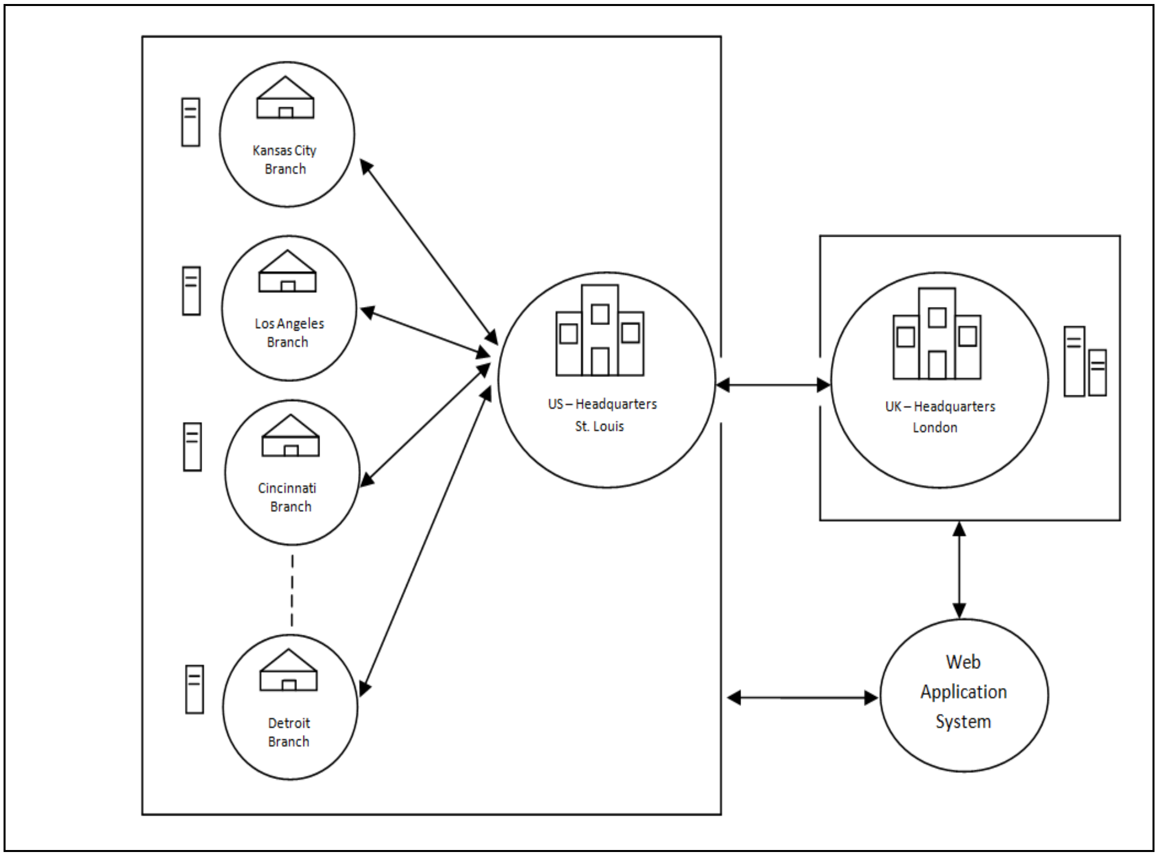

Figure 6.4. HSBS's Proposed Future Setup

The arrows in figure 6.4 represent the information exchange through a direct network connection.

Software Firm- HOBNOB INC (Hypothetical). The HOBNOB Inc is a provider of integrated business, technology and process solutions on a global delivery platform. It is a CMMi Level 5 certified Software Services Company headquartered in St Louis, Missouri. HOBNOB Inc has its presence in international locations like Europe and Asia. It has 55+ 'Centers of Excellence' that create customized solutions, no matter the domain involved .The US business unit of HOBNOB Inc Limited is one of the fastest growing companies in the North America. The units offer a 360 degree service portfolio spanning the entire IT life cycle. This includes Consulting, Business Solutions, System

Integration, Infrastructure and Application Management and Total Outsourcing services where they service all IT needs of a customer, end-to-end. HOBNOB Inc. employs around 50000 employees and has an excellent track record in terms of the quality of its products. Also, it has an excellent customer support.

After a rigorous discussion between the bank stakeholders and the core team of experienced developers from HOBNOB Inc, the following requirements are officially documented which both satisfy the customer needs and are feasible to be developed into an actual system. The old mainframe database systems might not be compatible with the some of the newer technologies on which a new e-banking application might be built, the old database system should be migrated to a new database system (like Oracle etc.). Both the headquarters i.e. one present in US and the other in UK should be securely connected. An e-banking web application should be developed, which is easy to use, consists of all the major features present below:

Transactional (e.g., performing a financial transaction such as an account to account transfer, paying a bill, wire transfer and applications apply for a loan, new account, etc.) , electronic bill presentment and payment – EBPP, funds transfer between a customer's own checking and savings accounts, or to another customer's account, Investment purchase or sale, loan applications and transactions, such as repayments of enrollments, non-transactional (e.g., online statements, check links, co-browsing, chat), bank statements, financial Institution, administration, support of multiple users having varying levels of authority, transaction approval process and wire transfer. Overall the system should have good documentation so that it is easy to maintain and the performance of the system should be robust.

The HSBS Corporation Bank gives HOBNOB Inc a probable deadline of 4-5 years to develop the complete set of system and services for them. This deadline is planned by the managerial team of HSBS Inc as per their expansion plans.

The strategy of development of this project by HOBNOB Inc. is developed as follows. The core team consists of experienced professionals from HOBNOB Inc who layout a plan for executing the set of business requirements asked from them by HSBS managers into technical requirements. They split the project in four phases.

Phase 1 (Deadline set around 18- 24 months)

Phase 1 consists of taking a new database system and customizing it according to the HSBS requirements. Main steps include designing, coding and testing suitable plug-ins that could help in migration of the data from the old mainframe legacy system to the new database system. It also includes developing two stand alone databases systems using the newer technology for both the headquarters in US and UK. This phase would include around 200 people during its development. Phase 1 is a very critical phase as a lot of highly confidential information will be transferred from the old system to the new system.

Phase 2 (Deadline set around 9 months)

By this time the bank would be moving to its new headquarters in UK. All the necessary networking protocols such as proper encryption/decryption schemes should be in place, so that the headquarter in UK could be connected with the US headquarter. This phase also includes incorporation of various security features in the system developed in phase 1 so as to keep the send/ receive of information secure. The phase would see an addition of 100 new people joining those already working in phase 1. This phase has to address all the network related security issues and make sure that the network is perfectly secure for all the bank operations.

Phase 3 (Deadline set around 12 months)

To design, code, test and deploy an e-banking web application. The web application should facilitate both the employees and the customers. It should properly authenticate between the customers and employees. Customers can use the e-banking application for all the general purposes like viewing their accounts, transferring money, viewing balance statements, and chatting with customer care etc.

The employees in addition to the features available for customers have certain added features accessible to them. These can be; viewing & updating records for multiple customers, processing loans for customers, refunding money to a customer's account by rolling back the transaction etc. This phase would see an addition of 150 new people to the people already working on previous two phases.

Phase 4 (Deadline set around 12 months)

This phase deals with maintaining the entire system developed during phase1, phase 2 and phase3. The maintenance would be undertaken for a contract time of 12 months. Some of the tasks that will be performed during this stage are as follows: Keeping a log of transactions that occur in the bank's system to check for possible problems. Checking the data flow in the system and making sure if it is right according to the various document specifications. Checking for inconsistencies in the system elements like database, e-banking web application etc. Handling customer complaints from the bank and resolving it successfully. Maintenance phase would only include 100 people working on the project as a part of the maintenance team.

The core team decides to follow the spiral model for software development process for all the three phases. The project would involve roughly 400 people during its entire development lifespan of 4 to 5 yrs. These people will be responsible for all the processes like software development process, networking issues, user-interface development etc and would work in synchronization with HSBS employees. All the necessary SQA (Software Quality Assurance) policy is laid down by the core team.

The budget for this project is decided to be around 10 million $ + maintenance costs. The budget is decided by discussion and analysis by the both the parties. HSBS agrees on paying 20$/hr for the entire workforce for the entire lifespan of the project.

6.4. SURVEY 1

The purpose of Survey 1 was to capture the stakeholders initial thoughts regarding the issue and their preferences of solution alternatives. What that meant was, how different stakeholders favored different alternatives and what was their rationale behind favoring those alternatives. The Survey 1 consisted of two simple questions. The first question asked them as to which alternative solution did they favor the most. The second question asked them about the reason for their belief. The results for Survey 1 were captured in period of one week and Survey 1 served as the base for analysis of different results achieved during the argumentation process.

Out of 25 stakeholders, 24 participated in Survey 1. Figure 6.5 shows the split of support for four different alternatives of the issue by the stakeholders.

In other words figure 6.5 also captures the vote distribution of different stakeholders in Survey 1. The pie chart shows the voting distribution for different alternatives according to the answers provided by them on the first question of the Survey 1. The chart clearly shows that 45.83% (11 in total) of stakeholders initially believe that a combination of 50% Software Quality Assurance and 50% Software Testing will be favorable for resolving the issue in a scenario which resembles the given hypothetical background information. 29.17% (7 in total) of stakeholders believe in a combination of 20% Software Testing and 80% Software Quality Assurance whereas 25% (6 in total) believe that a mix of 80% Software Testing and 20% Software Quality Assurance will be favorable for this situation.

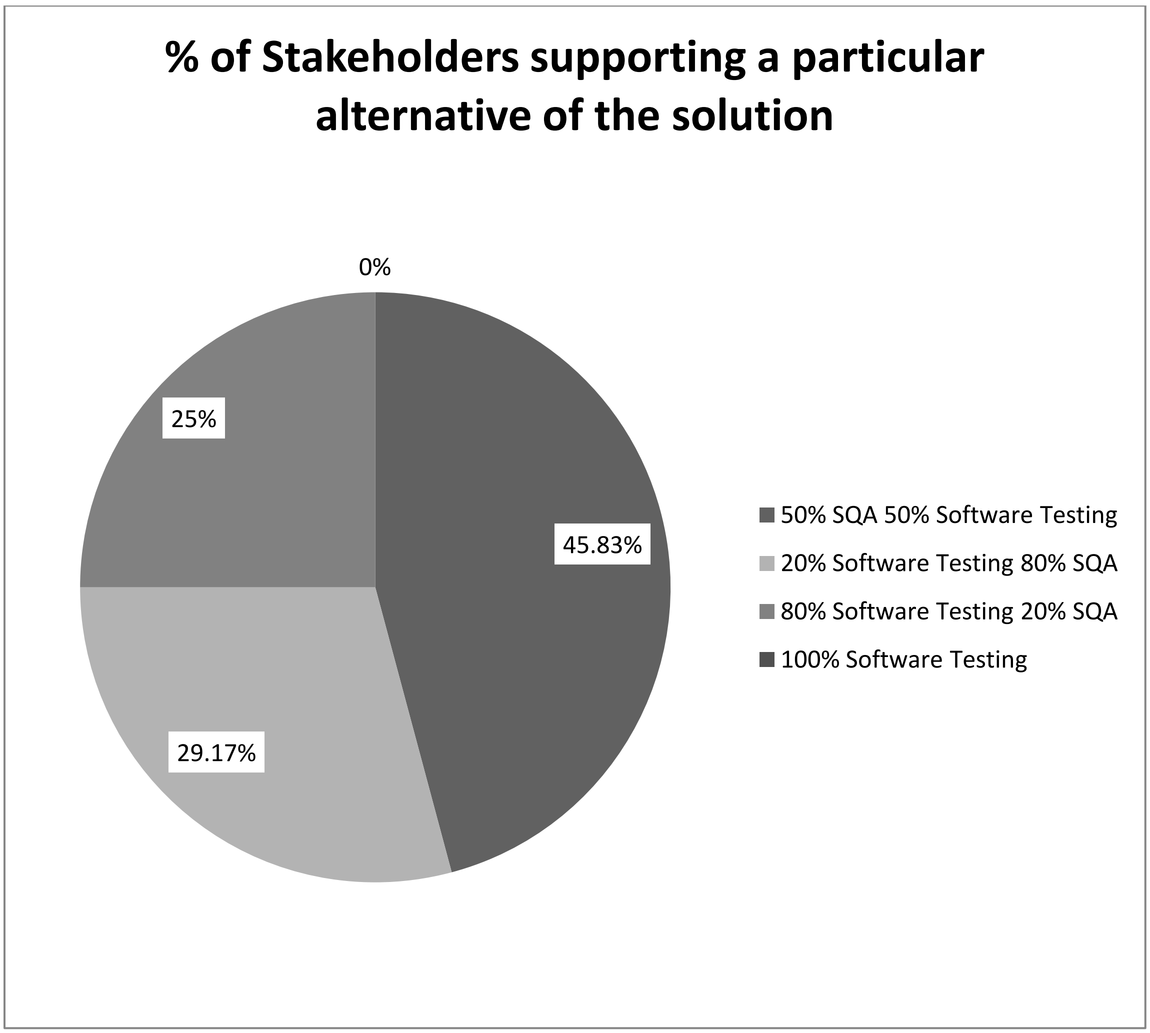

Figure 6.5. Vote Distribution in Survey 1

The last option i.e. 100% Software Testing is not supported by any of the stakeholders as a suitable technique for the hypothetical scenario.

Along with the data captured in question 1 of the Survey, the stakeholders were also asked the reasons of their support for a particular alternative. This reason served as their rationale for supporting their belief. The reason description demonstrates the different ways in which the need for that particular rationale was stated in the second question of Survey 1.These rationales were distinctly captured in a set of 8 different groups. Table 6.1 shows the list of reasons that broadly categorized the stakeholder's rationales along with the description of those rationales. The description specifies the context in which the reason has been stated in different survey inputs by the stakeholders.

This classification of rationales in groups later served as an excellent methodology for understanding the widening of the stakeholders thought process during the argumentation process and Survey 2.

Table 6.1. Stakeholders Reason and Reason Description

Reason ID	**Reason Name**	**Reason Description**
1	Need for a well defined Processes	• Since the project is a combination of large hardware and software systems, it requires a well planned design of execution steps. • The planning can be either in software architecture or hardware implementation. • It will also include a plan to apply the workforce regularly and in an optimized way during different stages of project lifecycle. • Planning is also required for successful migration from the legacy system to the new system and will ensure a good

Table 6.1. (Continued)

		quality system which will meet high standards. • Well defined processes also imply that the company should employ Total Quality Management. • A well planned system will reduce the error propagation rate and catch the errors in the early SDLC phases. • It will also make the system more flexible to allow future changes and scalable. • It will also lead to a good documentation.
2	Need for Efficient Testing & Reporting	• The project involves the banking industry which is a highly regulated industry and will probably only become more regulated. Because of this reports of testing and results will have to be extensive and retained for any regulatory body governing the industry.
3	Need for Reusability	• Use of previous System architectures with customization. For example could be very well used for designing e-banking applications
4	Need for Adhering to Customer Requirements	• This tells how the customer requirements are met and dealt with. These customer requirements can be either direct or

Table 6.1. (Continued)

		indirect. • It can help achieve high customer satisfaction. • It can reveal misunderstood user requirements
5	Need for Having high security	• Prevention against unauthorized attacks and loss of important data
6	Need for Removing Defects	• Removing bugs is the primary concern
7	Need for Effort Conservation	• Saving money, time, and man-hours. • Meeting deadline is important. For that reason even testing can be cut short. • Some think more SQA can conserve effort while some think less SQA can do it. • Maintenance costs can be minimized
8	Dependency on Scope of the Project	• If the scope of the project is big, it might need more SQA, but a small scoped project needs less SQA and may be more testing. • Also a high quality system will cost more to be maintained than a low quality system. • It also highlights the style in which the project is being designed, which is more inlined with the waterfall model.

Table 6.1. (Continued)

		• Because of a large scope of the project at hand, 100% testing is not feasible

As already stated; Table 6.1 captures the reasons that different stakeholders gave during their participation in Survey 1. Table 6.2 shows the number of hits encountered for each reason in Survey 1. Number of hits signifies the number of people supporting or stating the rationale as mentioned in the reason name of table 6.1 during the course of Survey 1.

Table 6.2. Number of Supports for Rationale during Survey 1

Reason ID	**Reason Name**	**Hits in Survey 1**
1	Need for a well defined Processes	2
2	Need for Efficient Testing & Reporting	1
3	Need for Reusability	2
4	Need for Adhering to Customer Requirements	3
5	Need for Having high security	6
6	Need for Removing Defects	13
7	Need for Effort Conservation	10
8	Dependency on Scope of the Project	1

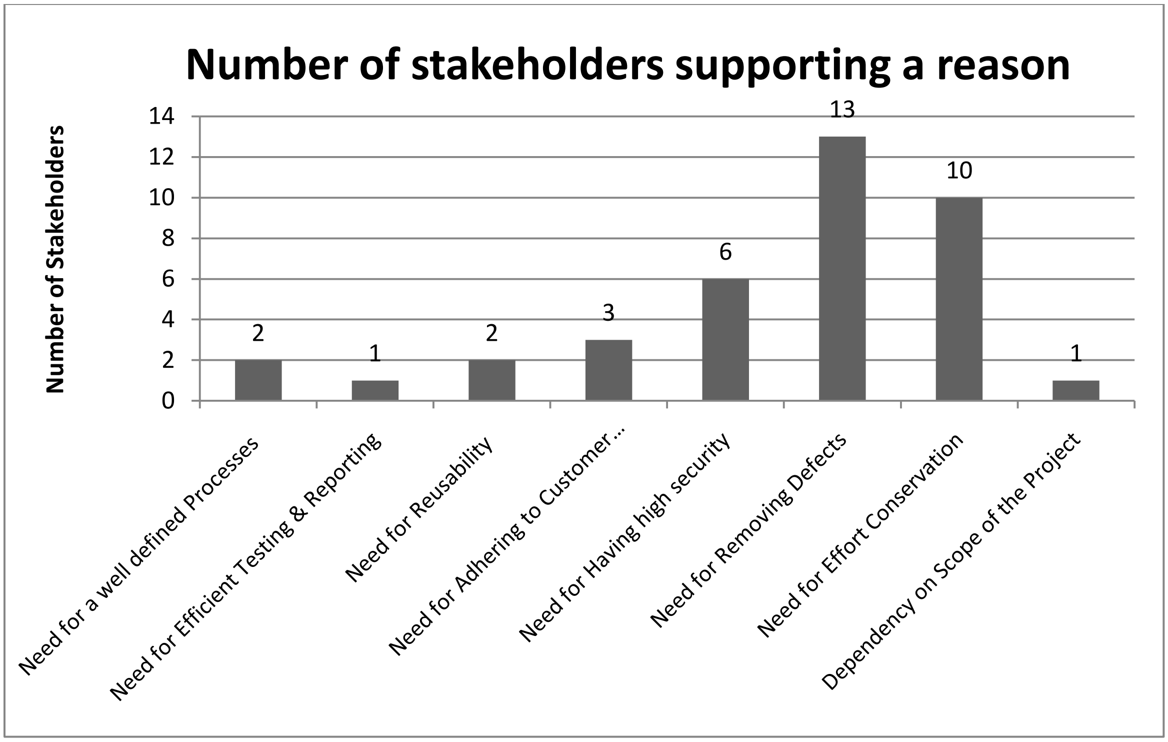

Figure 6.6 . Stakeholders Support for a Particular Rationale in Survey 1

As we can clearly see in figure 6.6, the reason that is most supported by the stakeholders is the "Need for Removing Defects". This analysis reflects the stakeholder's initial response to the case study and selection of a suitable alternative for the distribution of percentage of effort between software quality assurance and software testing.

6.5. THE ARGUMENTATION PROCESS

After participating in the Survey 1, the stakeholders were asked to participate in the Argumentation process. The attributes of the argumentation system were as follows:

Issue-The issue specifies the problem statement about which the stakeholders will provide their arguments and reach out a consensus. The issue for our case study was "To determine percentage of Effort distribution between Software Quality Assurance and Software Testing for the given hypothetical scenario".

Criteria- The criteria specify the domain under which the current issue was debated. Although there could have been many criterions for consideration, 4 specific criteria were taken so as to limit the discussion to a considerable length with regard to the time available for the whole process. The set of four criteria were “Making the whole system highly secure”,” Satisfying the requirements of the customer”, “Meeting the project Deadline”, and “Reducing the maintenance costs”. Each criterion was assigned a weight on a scale of 0 to 1. The process for assigning the weights to the criteria is explained in the following sections.

Alternative- Alternatives specify the possible alternative solutions which could be discussed for resolving the issue. The alternatives identified for this case study were “100% effort on Software Testing”, “80% effort on Software Testing and 20% effort on Software Quality Assurance”, “50% effort on Software Testing and 50% effort on Software Quality Assurance”, “20% effort on Software Testing and 80% effort on Software Quality Assurance”. The term “effort” mentioned here could be in terms of the man-hours or budget allocated.

Students were given a time period of three weeks to provide their inputs using the argumentation process. Every argument carried with it a weight which represented its degree of attack or support for a particular alternative or an argument. The weight for each argument was given by the stakeholders themselves. To make students better understand the scale of the weight values, they were given a reference scale which could be used for assigning weighs to the arguments. The degree of attack was represented from a scale value of -0.1 to -1.0 and the degree of support was represented form 0.1 to 1.0. The value of 0 was considered indecisive. Table 6.3 shows the reference scale that was given to the students for assigning weights to the arguments.

Table 6.3 . Reference Scales for Argumentation Weights

Attribute	**Weight Value**
Strong Support	0.7 to 1.0
Medium Support	0.4 to 0.6
Weak Support	0.1 to 0.3
Indecisive	0

Table 6.3 . (Continued)

Weak Attack	-0.1 to -0.3
Medium Attack	-0.4 to -0.6
Strong Attack	-0.7 to -1.0

Along with the weights, the arguments also contained a priority value. The priority value in our system is a value between 0.1 and 0.9 [11].The priority value demonstrated the preference a stakeholder can have over other stakeholders. For the purpose of this case study, all students were given an equal priority of "0.5".

6.5.1. Assigning the Criteria Weights. The criteria weights were calculated using the Analytic Hierarchy Process and the values were put in the system against their respective criterion. Table 6.4 present the scale for comparing the values of one criterion with respect to another, table 6.5 show the actual comparison of each criterion value with respect to another, table 6.6 shows the normalized value for each criterion.

Table 6.4 . Scale for Comparison Between Criteria

Value a_{ij}	**Comparison Description**
1	Criteria i and j are of equal importance
3	Criteria i is weakly more important than j
5	Criteria i is strongly more important than j
7	Criteria i is very strongly more important than j
9	Criteria i is absolutely more important than j

Table 6.5 . Score Based Comparison Between Different Criterions

	Making the System Highly Secure	**Satisfying Customer Requirements**	**Meeting Project Deadline**	**Reducing Maintenance Costs**
Making the System Highly Secure	1	3	5	3
Satisfying Customer Requirements	1/3	1	5	3
Meeting Project Deadline	1/5	1/5	1	1/3
Reducing Maintenance Costs	1/3	1/3	3	1

Table 6.6 . Normalized Score of Criteria Weights

	Making the System Highly Secure	**Satisfying Customer Requirements**	**Meeting Project Deadline**	**Reducing Maintenance Costs**
Making the System Highly Secure	0.538	0.66	0.357	0.409
Satisfying Customer Requirements	0.177	0.230	0.357	0.409
Meeting Project Deadline	0.108	0.044	0.071	0.045
Reducing Maintenance Costs	0.177	0.066	0.214	0.136

The final values of each criterion were calculated by taking the average of the values across each of the four rows and were as follows:

Making the system highly secure – 0.49

Satisfying Customer Requirements – 0.29

Meeting Project Deadline – 0.07

Reducing the maintenance costs – 0.15

All the arguments which were provided by the stakeholders were structured in a hierarchy using the argumentation system. Every stakeholder provided a rich set of arguments which played a significant role in the argumentation process and further directed the decision making process. All the arguments were thoroughly analyzed and the arguments were categorized in a group of three quality levels namely “H”, “M”, “L” signifying High, Medium, Low .Figure 6.7 shows the distribution of arguments between different stakeholders along with their classification in the stated three quality levels. “Relevance to the case study” served as a principle for classifying an argument into any one of the three quality levels. Therefore, a high quality argument means its high relevance with the case study whereas a low quality argument means a low relevance to the case study. A medium quality argument represents a medium level relevance to the case study. Table 6.7 shows the distribution of arguments in the argumentation system with respect to their quality levels.

Table 6.7 . Distribution of Arguments With Respect to Their Quality Levels

Quality Level of Arguments	Percentage of Arguments
H	29.75
M	20.97
L	49.28

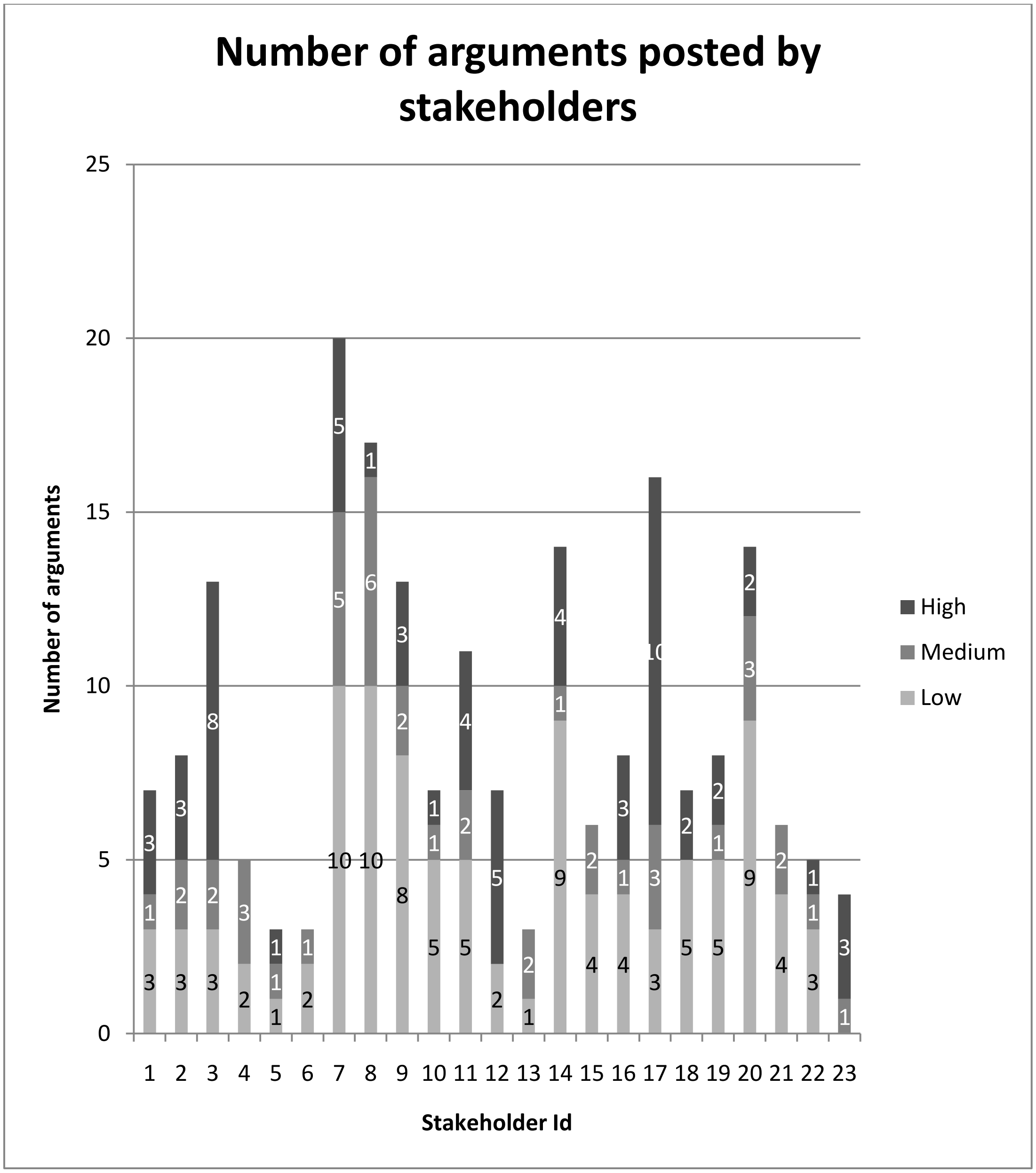

Figure 6.7 . Quality and Count of Arguments Posted by Different Stakeholders

The relevance to the case study was necessary as it was important that the discussion be strictly on facts rather than being a general discussion. Figure 6.8 demonstrates the rate with which the argumentation tree grew

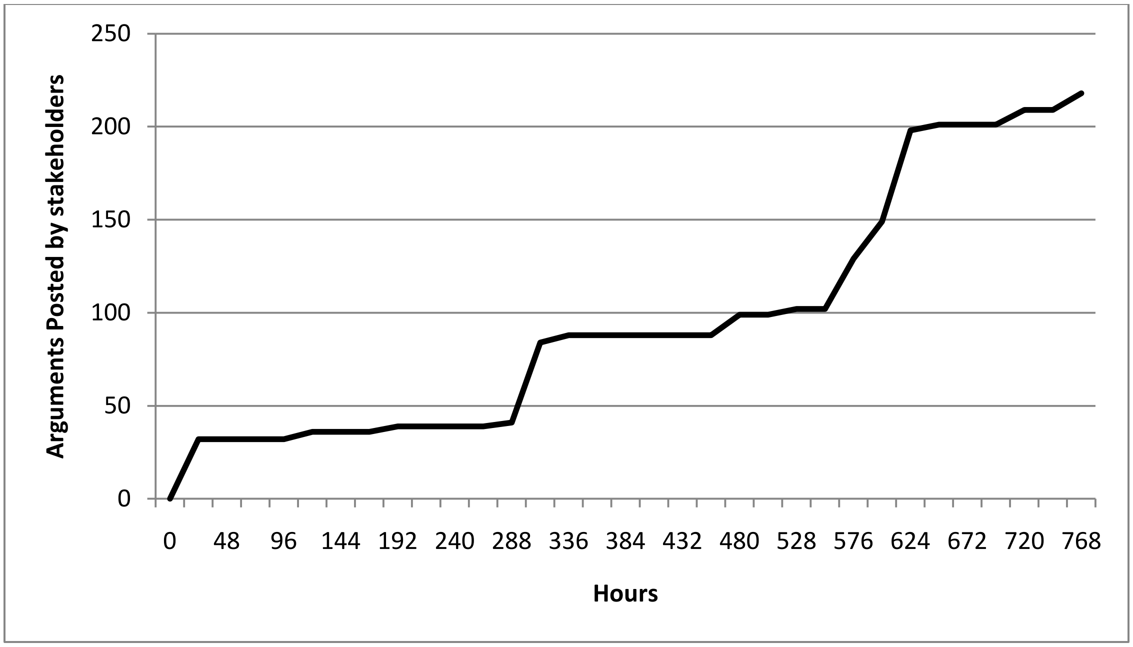

Figure 6.8 . Rate of Growth of the Argumentation Tree

The x-axis are the hours starting from the time the students were given access to the argumentation system. It was observed that the arguments were provided by the students in phases and there were times when there was very little activity in the argumentation process. There were sharp increases in the growth in number of arguments within short period of time which are easily noticeable on the graph in figure 6.8.

The arguments were spread in different levels during the argumentation process. The normal hierarchy of the system is shown in figure 6.9 consists of a node called *Project* which specifies the collection of issues to be argued upon. The *Issue* represents the decision issue that is to be resolved using argumentation. The issue contains *Criteria* nodes which are typically the four criteria for the current empirical study and as specified earlier. Each criteria node further contains the four *Position* nodes which are nothing but the alternatives to the issue at hand. The *Argument* node is present either under a position node or under another argument node. The *Evidence* node is present under the argument nodes and they can also be present under another evidence node.

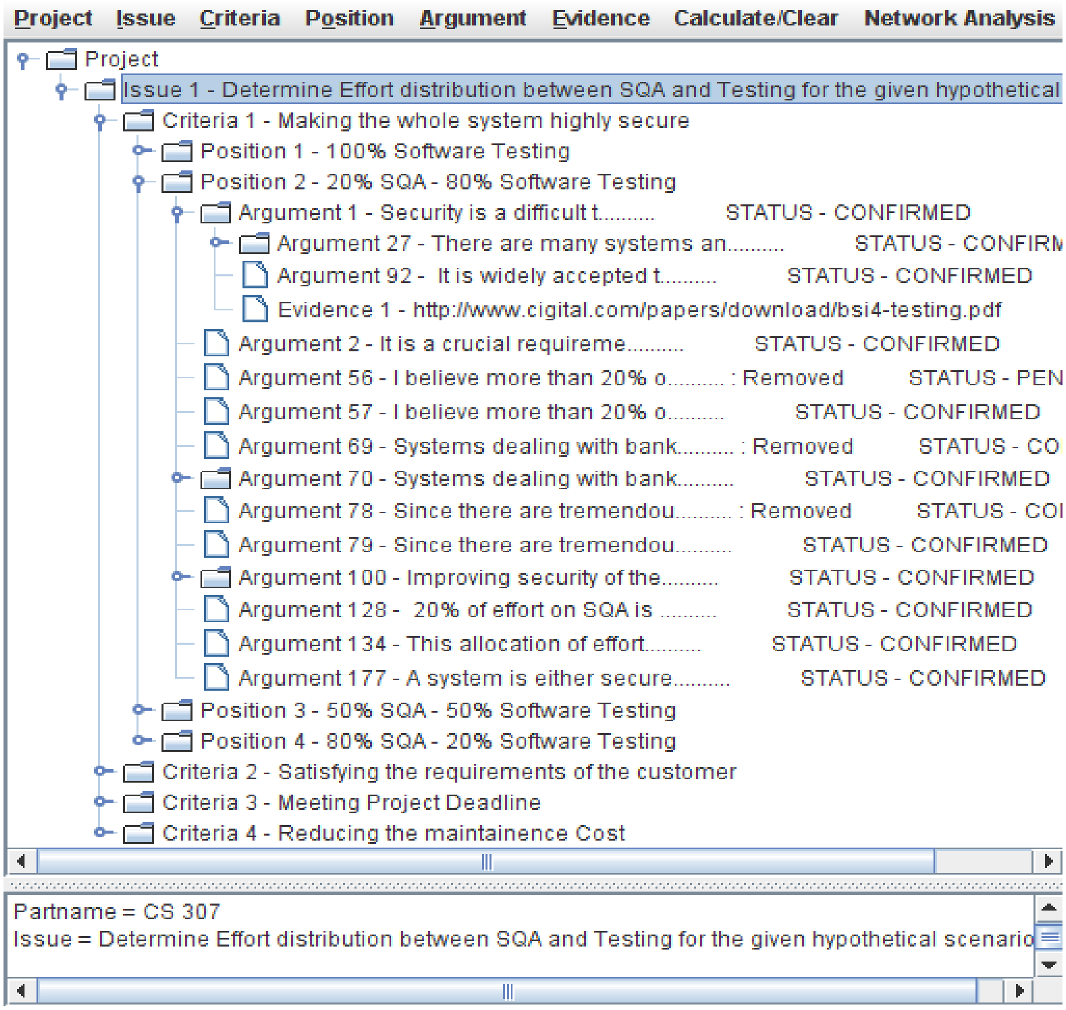

Figure 6.9 . The Hierarchy of Nodes in the Argumentation System

As we can clearly see, various arguments are present at different levels of hierarchy. Taking the position node as the top node or the node with level zero, table 6.8 shows the depth of the argumentation with the number of arguments present at that depth.

Table 6.8 . Presence of Arguments at Different Levels of Argumentation Tree

Depth of the Argumentation Tree	**Percentage of arguments**	**Percentage of evidences**
1	68.62%	0
2	25.98%	83.33%

Table 6.8. (Continued)

3	5.39%	11.11%
4	0	5.55%

The majority of arguments collected were present at the node immediately below the position node, which meant that they were directly supporting or attacking a position. This accounted for nearly 68% of the arguments for the given case study. Majority of arguments were placed in this level because each stakeholder had a view which corresponded to his support or attack for an alternative. It was after this step that the next level of argumentation took place in which nearly 25 % of the arguments were placed in direct contact with the arguments at level 1. These arguments supported or attacked the arguments at level1. This was a result of the ongoing debate between various stakeholders and their conformance and non conformance to each other's views and opinions. We also observed arguments at level 3 which were placed directly below the arguments at level 2 were either supporting or attacking the arguments at level 2 or acted as a reply to arguments at level 2 by the stakeholders.

Evidences are not present at level 1 because our system does not allow evidences to be added below an alternative. Majority of evidences (83%) are present at level 2. These evidences either attack or support the arguments at level 1. Arguments at level 2 are accompanied by evidences at level 3 and this accounts for nearly 11% of total evidences collected in the system. No further arguments were observed at level 4 but instead level 4 had presence of evidences in place of arguments.

After the arguments are structured, our system makes use of the fuzzy inference engine and fuzzy association matrix as in [10] and [11] to calculate the impact of indirect arguments on the position nodes. A performance score was obtained for every alternative criterion pair. The decision matrix representing the performance scores are displayed in table 6.10. Table 6.9 presents the number of attacking and supporting arguments for each criterion-alternative pair. The '+' sign represents the support for an alternative whereas the '-' sigh represents the attack. It can be noticed that the values in table 6.10 are directly related to the number of support and attack arguments represented in table 6.9. The number of supporting and attacking arguments can easily give us an idea of how the weights for each alternative-criterion pair differ in terms of the number of supporting and attacking arguments it contains.

Table 6.9. Decision Matrix with Number of Supporting and Attacking Arguments

	Making the System Highly Secure (weight-0.49)	**Satisfying Customer Requirements (weight-0.29)**	**Meeting Project Deadline (weight- 0.07)**	**Reducing Maintenance Costs (weight- 0.15)**
100% Software Testing	5(+), 14(-)	8(+), 9(-)	4(+), 7(-)	4(+), 8(-)
20% SQA- 80% Software Testing	10(+), 7(-)	7(+), 4(-)	7(+), 4(-)	7(+), 3(-)
50% SQA- 50% Software Testing	12(+), 3(-)	9(+), 1(-)	12(+), 1(-)	10(+)
80% SQA- 20% Software Testing	6(+), 6(-)	10(+), 3(-)	11(+), 2(-)	10(+), 9(-)

Table 6.10. Decision Matrix with Performance Scores

	Making the System Highly Secure (weight-0.49)	**Satisfying Customer Requirements (weight-0.29)**	**Meeting Project Deadline (weight- 0.07)**	**Reducing Maintenance Costs (weight- 0.15)**
100% Software Testing	-3.163	-0.784	-1.095	-2.08
20% SQA- 80% Software Testing	1.213	1.57	0.74	1.45
50% SQA- 50% Software Testing	2.7	2.2	2.65	3.36

Table 6.10. (Continued)

80% SQA- 20% Software Testing	0.186	1.545	2.07	1.095

After the scores were calculated using the fuzzy association matrix and fuzzy inference engine, the scores were put through the process of weighted summation as explained in section 5.4. The final aggregated scores were as follows:

100% Software Testing = -2.166

20% SQA- 80% Software Testing = 1.319

50% SQA- 50% Software Testing = 2.65

80% SQA- 20% Software Testing = 0.848

Hence, the argumentation process supported that the most favorable alternative after the discussions between the stakeholders was "50% SQA 50% Software Testing". The second most favored was "20% SQA and 80% Software Testing" and the third most favored was "80% SQA and 20% Software Testing". The second and third most favored alternatives were opposite to what was observed during the Survey 1. It clearly shows the effect of argumentation system in refining the thought process of the stakeholders in reaching a decision.

6.5.2. The Shift in Favorability of Different Alternatives. Figure 6.10 presents the change in the overall favorability of the four alternatives with respect to the increase in the number of arguments. Overall the third alternative i.e. "50% Software Testing and 50% SQA" was the most favored one with respect to the rest three, but initially, the behavior was somewhat different.

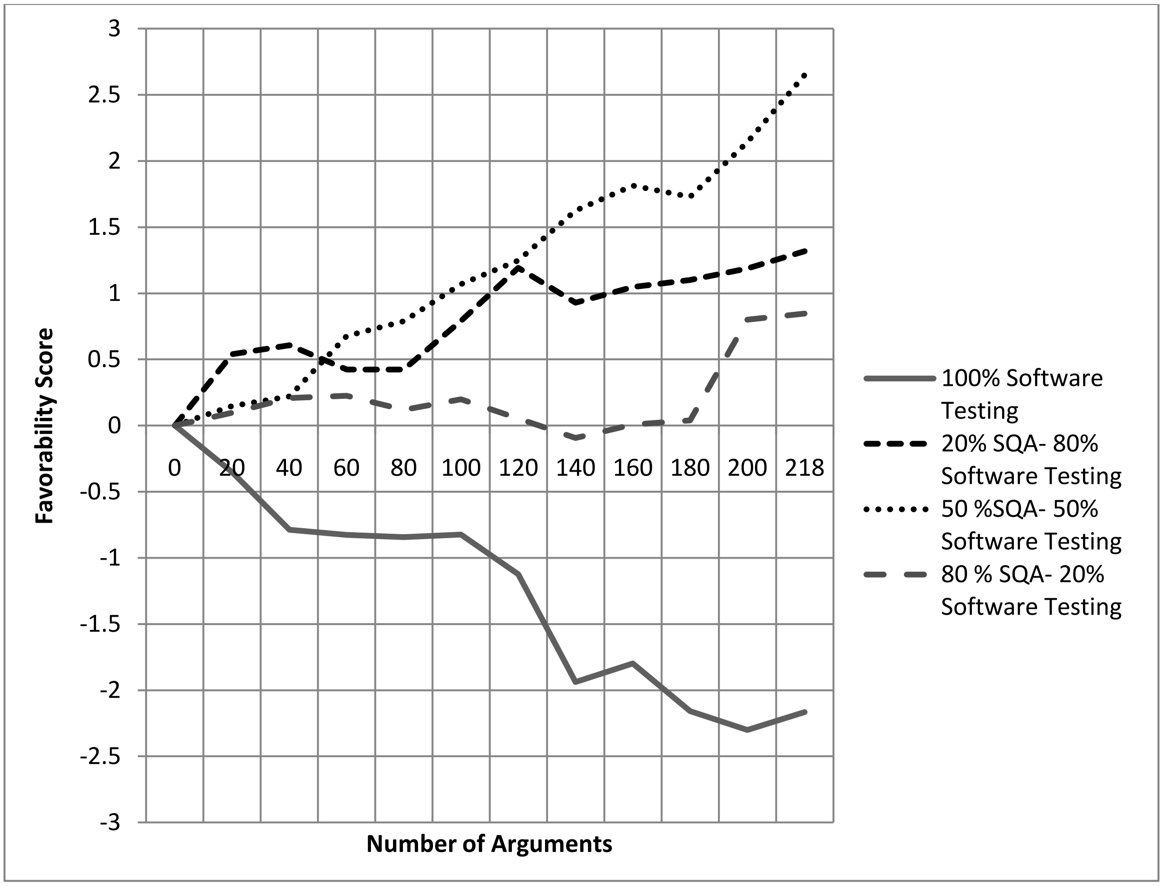

Figure 6.10 . The Change in the Most Favorable Alternative During Argumentation

It can be seen clearly that till 30 arguments, the second alternative i.e. "20%SQA and 80% Software Testing" had more favorability than the third alternative. Also till that time the fourth alternative which is "80%SQA and 20% Software Testing" had equal favorability as the third alternative. Till 50 arguments, the alternative 2 is still more favorable than alternative 3. Alternative 3 witnesses a steady increase in its favorability. Alternative 2 firstly grows positively and then slows down and then again picks up between argument number 80 and 120. Around argument 120 the second alternative is almost as favorable at the third alternative .Alternative fourth grows slowly during the argumentation process but witnesses a rise in its favorability between argument 150 and 200. The first alternative i.e. "100% Software testing" is always disliked and its favorability goes negatively during the entire process of argumentation.

The shift in the favorability factors of different alternatives shows us the movement of intelligence in the decision making domain. As compared to other approaches for decision making such as online web forums, emails, message boards etc. the argumentation process lets stakeholders analyze the arguments of other stakeholders and support or attack them with suitable weight values. This kind of shift in thought process can be easily seen in argumentation process and is mostly absent in some of the other discussed techniques.

6.5.3 Rationale covered in Argumentation. Similar to Table 6.2 which captures the number of times a particular rationale was covered by stakeholders during Survey 1, Table 6.11 reveals the number of times the same set of rationales was covered by different stakeholders during the argumentation process.

Table 6.11 . Number of hits for Rationale supported during Argumentation

Reason ID	**Reason Name**	**Hits in Survey 1**
1	Need for a well defined Processes	22
2	Need for Efficient Testing & Reporting	9
3	Need for Reusability	2
4	Need for Adhering to Customer Requirements	17
5	Need for Having high security	14
6	Need for Removing Defects	6
7	Need for Effort Conservation	18
8	Dependency on Scope of the Project	3

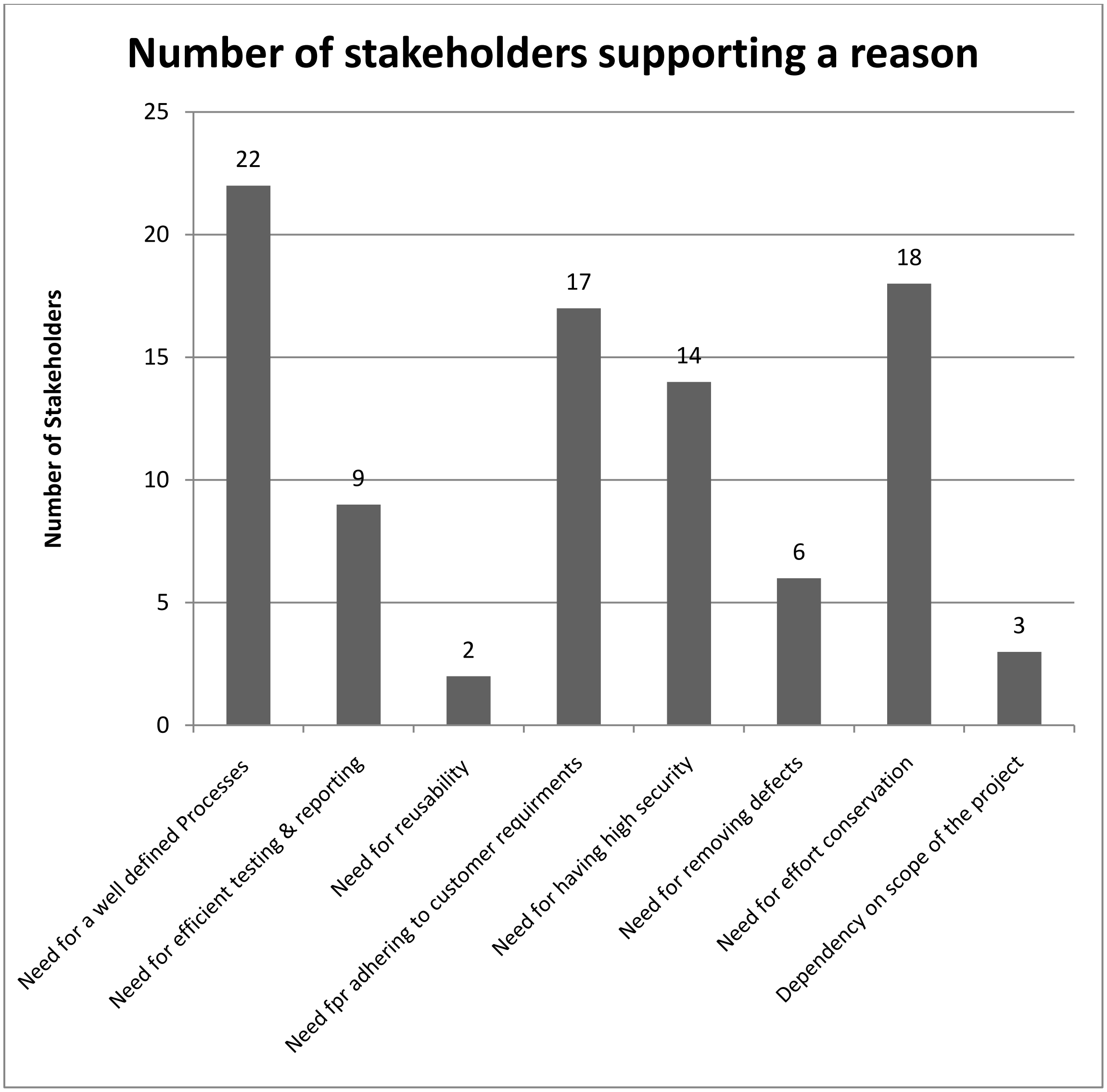

Figure 6.11 . Stakeholders Support for a Particular Rationale in Argumentation

Figure 6.11 presents another view of table 6.10.As we can see clearly in figure 6.11, the need for a well defined process is felt most among the stakeholders during the argumentation process. The next two are the need for effort conservation and the need for adhering to the customer requirements. The need for reusability is felt the least among the stakeholders.

So as to compare the difference in the number of stakeholders supporting a particular rationale from table 6.1 during the process of Survey 1 and argumentation, figure 6.12 is designed.

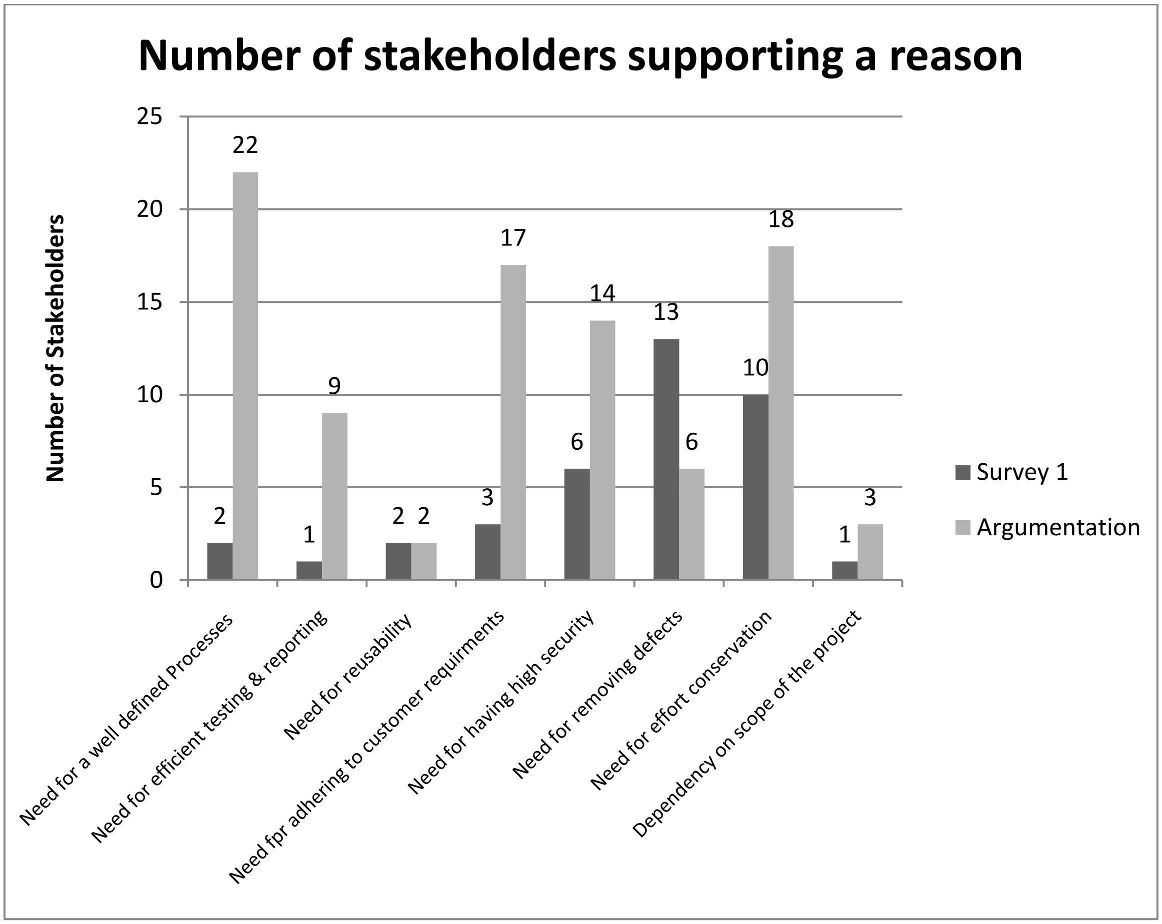

Figure 6.12 . Comparison Between Stakeholders Support for a Particular Rationale in Survey 1 and Argumentation

As can be observed from figure 6.12, the argumentation process enables stakeholders to have exposure to more number of rationales. Figure 6.12 shows that 7 out of 8 rationale show an increase in the number of stakeholders addressing them directly or indirectly. One rationale which is “need for removing defects” shows a decrease in the number of stakeholders support. This might be due to a change in the original belief of most of the stakeholders from Survey 1 after they have been through the argumentation process.

6.6. SURVEY 2

After the students participated in the argumentation process, they were again asked to participate in a second survey. The pre requisite for filling the second survey was that the students had to review the argumentation tree so as to make any inputs in the Survey 2 valid. The second survey was similar to the first survey, but in addition had two more questions. The first new question asked them about the percentage of arguments they reviewed and the second new question asked them if they had a change in their opinion from Survey 1 and if yes how. The motivation behind collecting data from Survey 2 was to check, how many stakeholders actually experienced a change in their opinion from Survey 1 and in why was this change caused. The chart in figure 6.13 was constructed based on inputs from question 1 of the second survey which simply tried to capture the opinions of the stakeholders after the argumentation process

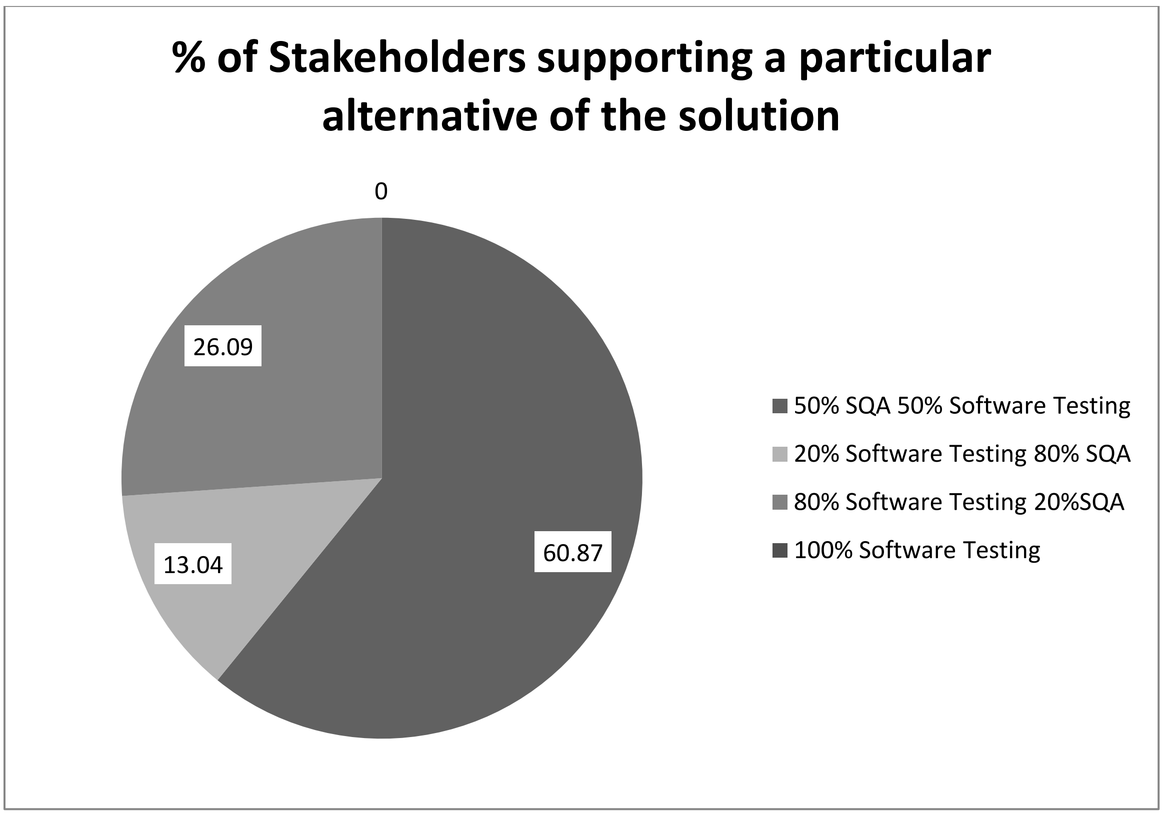

Figure 6.13. Vote Distribution in Survey 2

Figure 6.14 provides a graphical representation of how the distribution of votes varied from Survey 1 to Survey 2.

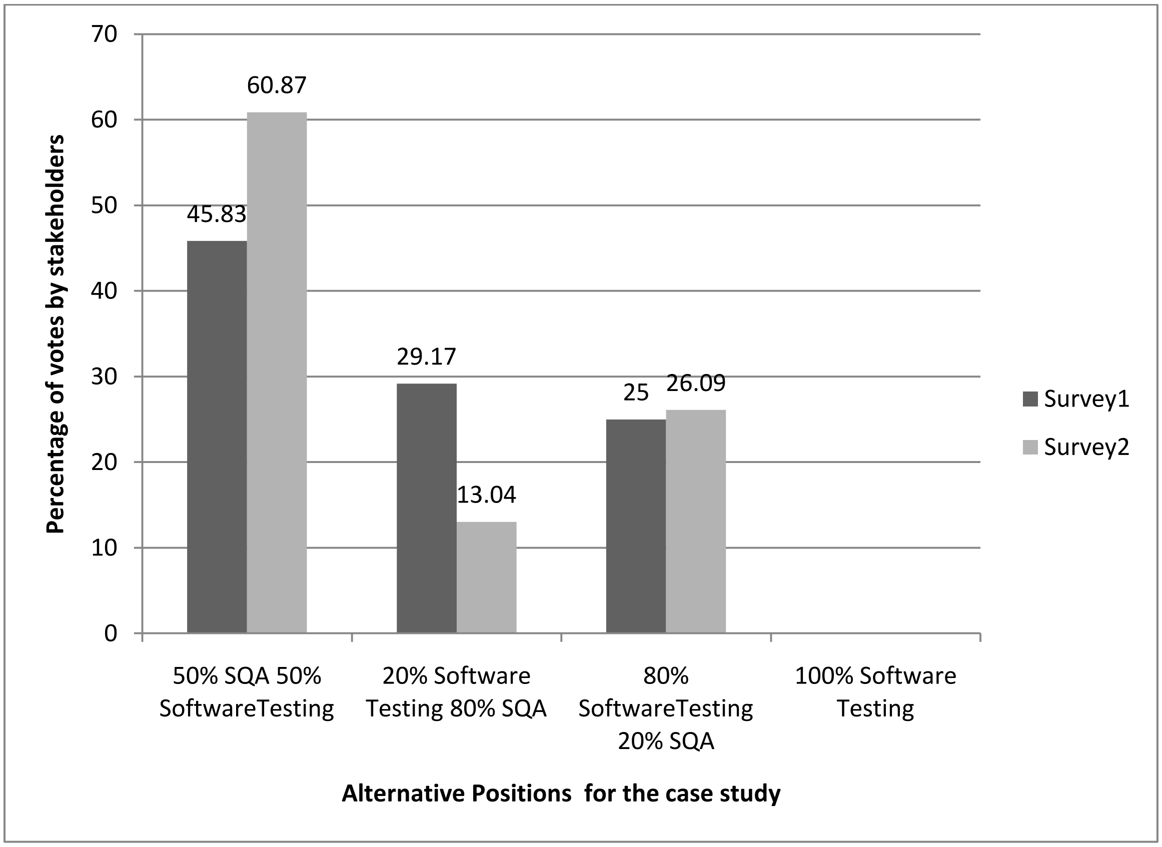

Figure 6.14. Comparison in Vote Distribution Between the Two Surveys

It can be seen from figure 6.14 that conformance to the alternative of “50% SQA 50% Software Testing” has increased since the argumentation process. We can also observe that there is change in positions in the second and third most favored alternatives among the stakeholders during the course of the two surveys .This is in direct correlation with what was observed during the results of the argumentation system. The argumentation system also showed that the second and third most favored alternative was opposite to what was observed in Survey 1. Table 6.12 presents the different types of transition in the opinions of stakeholders. This change was categorized in four classes i.e. “Opinion Changed”, “Opinion Reaffirmed”, “Opinion Weakened” and “Opinion Unchanged”.

Table 6.12 . Change in Opinion from Survey 1 to Survey 2

	Opinion Changed	Opinion Reaffirmed	Opinion Weakened	Opinion Unchanged
Number of Stakeholders	8	6	2	5

As can be seen in table 6.12, there was a change in the opinions of most number of stakeholders i.e. 8. Other than that 6 stakeholders felt that there opinion had weakened and 2 felt that there opinion had been strengthened.

Figure 6.15 tries to capture the rationales that were supported as a part of Survey 2 by the stakeholders.

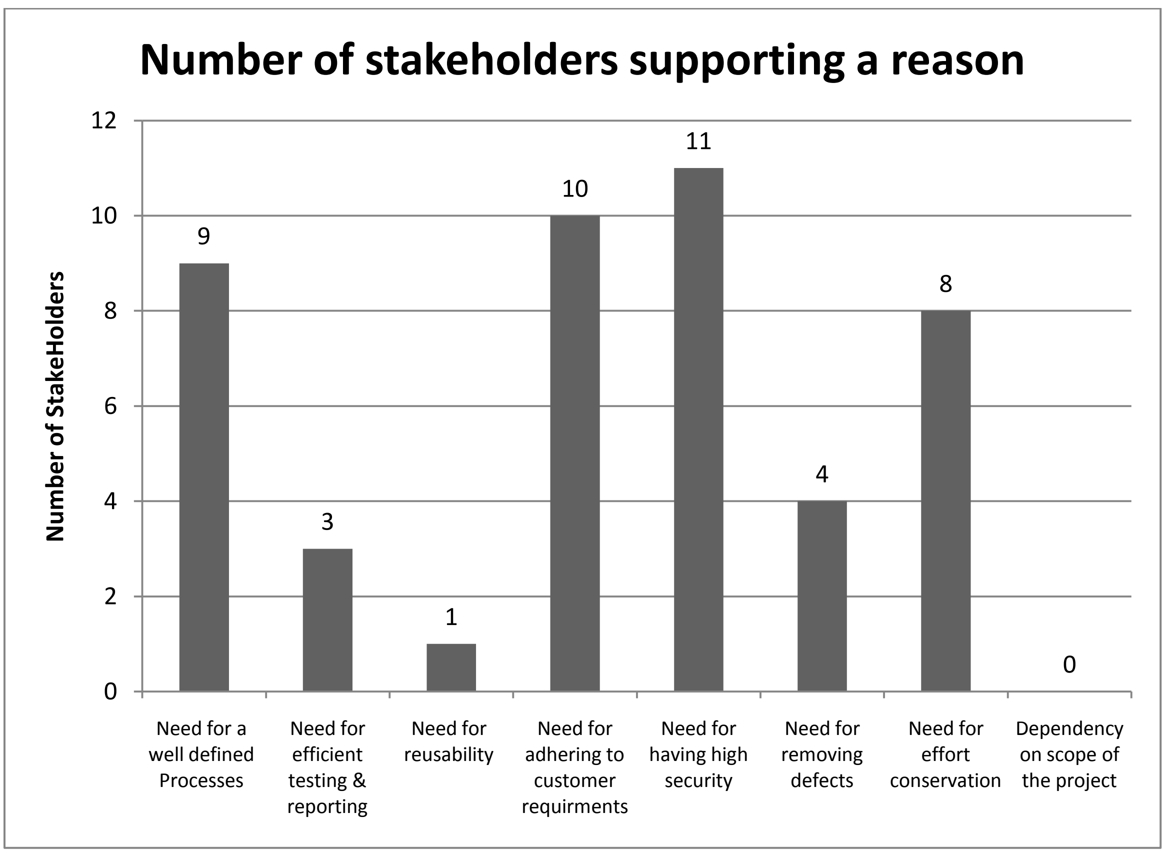

Figure 6.15 . Stakeholders Support for a Particular Rationale in Survey 2

Figure 6.16 tries to demonstrate the difference in the support for a particular rationale in table 6.1 between the three processes of Survey 1, Argumentation and Survey 2.

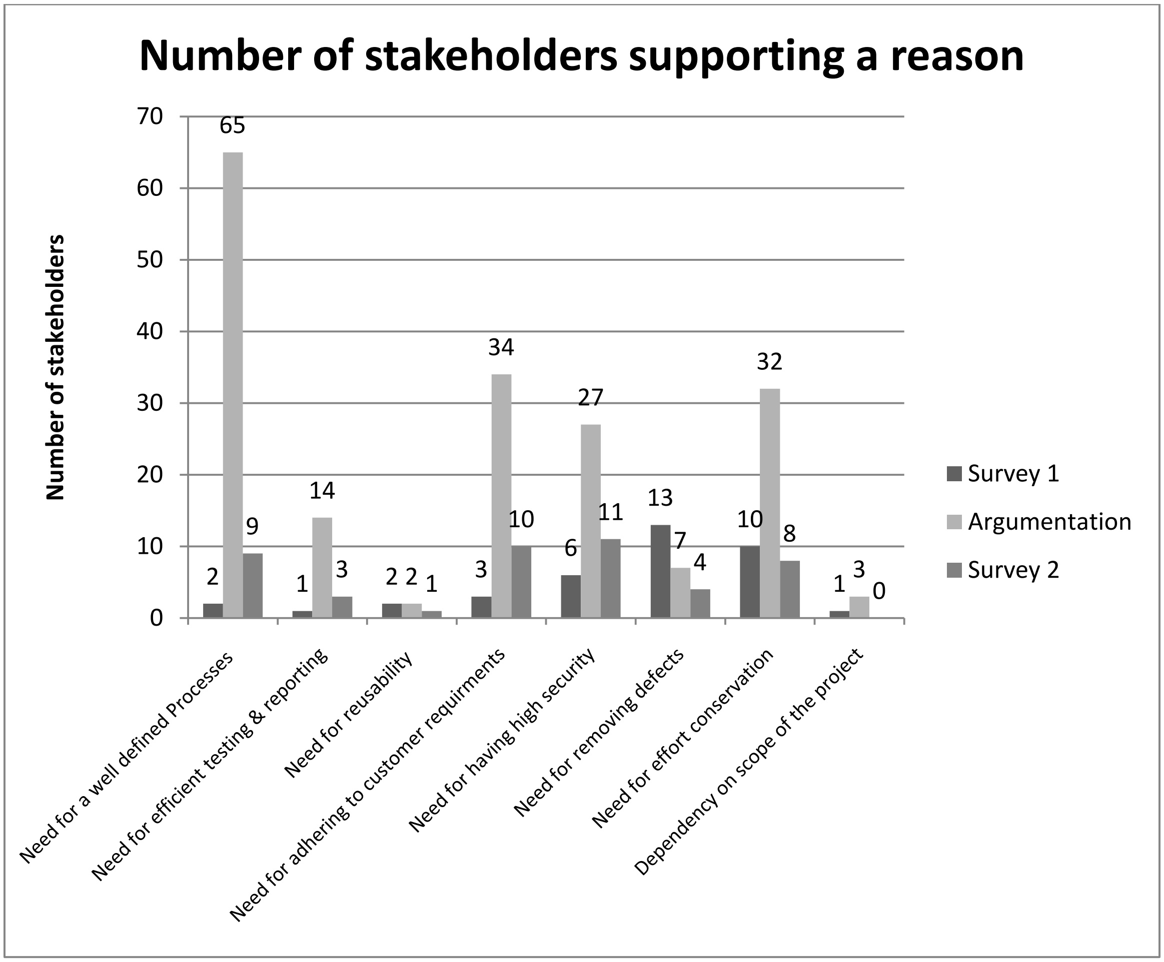

Figure 6.16 . Comparison Between Stakeholders Support for a Particular Rationale

While comparing the coverage of rationales during the three processes i.e. Survey 1, argumentation, and Survey 2, it is observed as in figure 6.16 that the argumentation process enables the stakeholders to gather the thought process of other stakeholders. This makes most of rationales to be available to most of the stakeholders, which enables in reaching a decision quickly. Information available to one stakeholder is easily made available to any other stakeholder who can then use that information to change his views and post an argument which could further help other stakeholders in changing their views.

Figure 6.16 also demonstrates the fact that, because of the argumentation process involved as a middle process for this study, there were more stakeholders supporting a particular rationale in Survey 2 than Survey 1. This was evident in 4 out of 8 rationales. This clearly hinted towards the validity of the argumentation tool to be an effective mechanism in aiding decision making in a collaborative environment.

7. CONCLUSION

The intelligent argumentation method and system can be used to assess performance scores in MCDM when they are controversial based on this study. The argumentation system was not only able to help us evaluate the performance scores for all the alternative-criteria pairs but also help produce a change in the stakeholder's views and opinions based on the fact that there was change in the second and third most favored alternative as well as a 15% increase in support for the most favored alternative from Survey 1 to Survey 2. The statistics clearly shows that the intelligent argumentation system is a handy tool in evaluating the performance scores in a MCDM framework and also aids in converging to a decision more rapidly. It also shows how knowledge of one stakeholder leads to change of views and opinions through effective argumentation and decision rationale capturing in collaborative decision making.

BIBLIOGRAPHY

[1] L. P. Amgoud. "Using Arguments for making and explaining decisions". *Artificial Intelligence 173* , 413-436, (2009)

[2] A. Bochman, "Collective Argumentation," *Proceedings of the Workshop on Non-Monotonic Reasoning*, (2002)

[3] G. R. Buyukozkan."Evaluation of software development projects using a fuzzy multi-criteria decision approach". *Mathematics and Computers in Simualtion 77* , 464-475, (2008)

[4] M. T. Chen. "Fuzzy MCDM Approach to Select Service Provider". *The IEEE International Conference on Fuzzy*, (2003)

[5] J. Conklin, and M. Begeman. “gIBIS: A Hypertext Tool for Exploratory Policy Discussion,” *Transactions on Office Information Systems*, 6(4): 303– 331, (1988)

[6] B. P. Duarte. "Developing a projects evaluation system based on multiple attribute value theroy". *Computer & Operations Research 33* , 1488-1504, (2006)

[7] E. G. Forman. "The Analytic Hierarchy Process- An Exposition". *OR CHRONICLE* , (1999)

[8] M. Lease, and J. L. Lively. “Using an Issue-Based Hypertext System to Capture Software Life-Cycle Process,” *Hypermedia*, 2(1), pp. 34 – 45, (1990)

[9] Peide Liu. "Evaluation Model of Customer Satisfaction of B2CE Commerce Based on Combination of Linguistic Variables and Fuzzy Triangular Numbers". *Eight ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel Distributed Computing, (pp* 450-454), (2007)

[10] X. F. Liu. "Management of an Intelligent Argumentation Network for a Web-Based Collaborative Engineering Design Environment". *Proceedings of the 2007 IEEE International Symposium on Collaborative Technologies and Systems,(CTS 2007), Orlando, Florida* , (May 21-25, 2007)

[11] X. F. Liu. "An Internet Based Intelligent Argumentation System for Collaborative Engineering Design". *Proceedings of the 2006 IEEE International Symposium on Collaborative Technologies and Systems* (pp. 318-325). Las Vegas, Nevada, (2006)

[12] T. M. "A subjective assessment of alternative mission architectures for the human exploration of Mars at NASA using multicriteria decision making". *Computer and Operations Research* , 1147-1164, (June 2004)

[13] A. N. Monireh. "Fuzzy Decision Making based on Relationship Analysis between Criteria". *Annual Meeting of the North American Fuzzy Information Processing Society* , (2005)

[14] N. Papadias. "HERMES: Supporting Argumentative Discourse in Multi Agent Decision Making". *Proceedings of the 15th National Conference on Artifical Intelligence (AAAI-98)* (pp. 827-832). Madison, WI: AAAI/MIT Press, (1998a)

[15] E. B. Triantaphyllou. "The Impact of Aggregating Benefit and Cost Criteria in Four MCDA Methods". *IEEE Transactions on Engineering Management, Vol 52, No 2* , (May 2005)

[16] S. H. Tsaur. "The Evaluation of airline service quality by fuzzy MCDM". *Tourism Management* , 107-115, (2002)

[17] T. D. Wang. "Developing a fuzzy TOPSIS approach based on subjective weights and objective weights". *Expert Systems with Applications* , 8980-8985, (2009)

[18] L. A. Zadeh. "Fuzzy Sets". *Information and Control 8* , 338-353, (1965)

VITA

Rubal Wanchoo was born on June 11, 1986 in New Delhi, India. He received his B.E. degree in Computer Science from Guru Jambheshwar University in 2008. He has been a graduate student in the Computer Science Department at Missouri University of Science and Technology since August 2008 and worked as a Graduate Research Assistant under Dr. Frank Liu from August 2009 to May 2010. He received his Master's in Computer Science at Missouri University of Science and Technology in August 2010.