TAES-COTS: Thorough Approach for Evaluation & Selection of COTS Products

Kashif Basir

2010-NUST-MS PhD-CSE-18

Supervisor

Dr. Aasia Khanum

DEPARTMENT OF COMPUTER ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY (NUST)

ISLAMABAD

AUGUST, 2014

TAES-COTS: Thorough Approach for Evaluation & Selection of COTS Products

Kashif Basir

2010-NUST-MS PhD-CSE-18

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Computer Software Engineering

Thesis Supervisor

Dr. Aasia Khanum

Thesis Supervisor's Signature: ____

DEPARTMENT OF COMPUTER ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY (NUST)

ISLAMABAD

AUGUST, 2014

APPROVAL CERTIFICATE

Certified that the contents of thesis document titled: ***TAES-COTS: Thorough Approach for Evaluation & Selection of COTS Products*** submitted by: **Mr. Kashif Basir** have been found satisfactory for the requirement of degree.

Supervisor: **Dr. Aasia Khanum**

Committee Member 1: **Dr. Farooque Azam**

Committee Member 2: **Dr. Saad Rehman**

Committee Member 3: **Dr. Arslan Shaukat**

External Member: **Dr. Sajid Anwar**

DECLARATION

I certify that this research work titled ***TAES-COTS: Thorough Approach for Evaluation & Selection of COTS Products*** is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources it has been properly acknowledged / referred.

Signature of Student

Kashif Basir

2010-NUST-MS PhD-CSE (E)-18

LANGUAGE CORRECTNESS CERTIFICATE

This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the university.

Signature of Student

Kashif Basir

2010-NUST-MS PhD-CSE (E)-18

Signature of Supervisor

COPYRIGHT STATEMENT

DEDICATION

I dedicate this thesis to my parents, family, friends and teachers.

ACKNOWLEDGEMENTS

In the name of Allah, the Beneficent, the Merciful. I am extremely thankful to Allah almighty for bestowing infinite blessings upon me and providing me with an opportunity to study at the most prestigious institute of Pakistan.

I am thankful to my parents for their continuous love, support and prayers. I would like to express my sincere gratitude to all my family members and friends for their encouragement.

I am very grateful to my supervisor Dr. Aasia Khanum for her guidance, suggestions, patience and motivation throughout the thesis. It was only her guidance that made it possible to complete this research.

I would like to thank Dr. Farooque Azam, Dr. Saad Rehman, Dr. Arslan Shaukat and Dr Sajid Anwar for being on my thesis guidance and evaluation committee. I am also thankful to them and Dr Shoab A. Khan for their insightful comments and suggestions.

I appreciate all the people who in any way assisted me in carrying out my research work.

ABSTRACT

Traditional software development process is gradually being replaced by Commercial Off-the-shelf (COTS) based software development (CBSD). The reason for this replacement is the availability of ready-made software packages. There are numerous benefits to the use of packaged software, reduced cost and less development time being few. The selection of only the most suitable COTS software product is an indispensable task in CBSD that should be done circumspectly. The success of such a development depends on the evaluation & selection methodology being followed. A perfect evaluation approach is expected to separate the wheat from the chaff. Various evaluation approaches have been proposed over the last few decades which rely on different criteria for singling out favorable contenders but the absence of a formal method indicates that this area of research is still miles away from its heyday.

Thorough approach for evaluation and selection of COTS products (TAES-COTS) proposed in this thesis is an approach that evaluates COTS products based on progressive filtration. It is a five step approach that evaluates products against different criteria at three different stages. It evaluates COTS software based on both functional and non-functional requirements and assists the evaluators in choosing the optimal software product in times of uncertainty. Effort has been made to come up with an approach that overcomes the limitations found in other approaches with the objective of being thorough enough to cover all aspects of COTS software selection and at the same time being simple enough to be practically implementable. To save the evaluators some time, this approach is accompanied by a prototype tool "COTSEVAL" that helps the examiners conduct the Non-Functional Requirements (NFRs) based evaluation.

TABLE OF CONTENTS

List of Abbreviations

AHP	Analytic Hierarchy Process
BBN	Bayesian Belief Network
CAP	COTS Acquisition Process
CARE	COTS-Aware Requirements Engineering
CBSD	COTS based software development
CEP	Comparative Evaluation Process
COTS	Commercial Off-The-Shelf
CRE	COTS based on Requirements Engineering
DSS	Decision Support System
ERP	Enterprise Resource Planning
FPS	First Priority in selection
GUI	Graphical User Interface
IROTS	Idealize Recommendation Off-The-Shelf
MCDA	Multiple-Criteria Decision Analysis
MCDM	Multiple-Criteria Decision-Making
MiHOS	Mismatch Handling for COTS Selection
NFRs	Non-Functional Requirements
OTSO	Off-The-Shelf-Option
PECA	Plan, Establish, Collect, Analyze

SPS	Second Priority in selection
STACE	Social-Technical Approach to COTS Evaluation
TAES-COTS	Thorough approach for evaluation and selection of COTS products
UnHOS	Uncertainty Handling in COTS Selection

List of Figures

List of Tables

Chapter 1 Introduction

This chapter consists of a brief introduction about various terminologies and concepts that are a part of this research work. The objective of this chapter is to build: a basic understanding of the concepts and the purpose of this research work. This chapter is divided into various sections. COTS, its examples and its benefits are discussed in section 1.1. Section 1.2 discusses differences between a software development process that involves use of COTS products and traditional software development process. Section 1.3 contains a discussion about the COTS product selection. Section 1.4 briefly discusses non-functional requirements. In section 1.5, decision support system is described. Section 1.6 contains information about AHP. Rule-based reasoning is discussed in section 1.7. Section 1.8 comprises background, motivation and scope. Section 1.9 contains the problem statement and thesis outline is describes in section 1.9.

1.1 COTS

"COTS" stands for Commercial Off-The-Shelf. Two things need to be discussed over here:

- Commercial
- Off-the-shelf

Commercial means that software is available for the use in the market but the consumer has to pay a fee for its usage or acquisition. Off-the-shelf means that the user needs not to develop it since it is already developed and available. So a COTS product refers to software that is readily available but its users are charged an amount for it [1].

1.1.1 Examples of COTS

A number of software products can be considered COTS. It can be antivirus software, Enterprise resource planning (ERP) software, an operating system etc. The most common example is Microsoft Office.

1.1.2 Advantages of COTS

There are a number of advantages of using COTS products. They are cheaper as compared to custom built software systems. Since they are readily available, the time consumed in development is saved. Design or production bugs are fixed earlier as there are numerous users to provide feedback [2].

1.2 COTS Based and Traditional Software Development

Traditionally, software development consists of phases like:

- **Requirement:**
- **Design**
- **Implementation**
- **Testing**
- **Deployment**
- **Maintenance**

But the phases and their order vary from model to model followed for the development. Also the activities conducted in each phase may vary as well. On the other hand a COTS based software development might contain Phases such as:

- **Identification:** Identification phase normally consists of activities like Preparation, Requirement Gathering, Planning, Methodology Selection, Product searching etc.
- **Selection:** In this phase activities regarding selection of most suitable candidate from a list of available candidates are conducted e.g. evaluation.
- **Integration:** Activities like integration of multiple COTS products etc are carried out followed by documenting the process etc.

And just like traditional software development, activities may vary here as well [3].

1.3 COTS Selection

The main focus of our research is COTS product selection. A number of selection methodologies are discussed in detail in chapter 2. The COTS selection phase of COTS based software development is a process in itself and perhaps the most crucial and decisive one. The successful completion of the whole development process depends mostly if not solely on choosing the right COTS candidate [4]. It is not possible to choose a suitable product in absence of a proper selection mechanism. A number of techniques have been proposed over the last few decades which rely on different criteria for singling out favorable contenders. The criteria used for this might be:

- **Functional Requirements**
- **Non-Functional Requirements (NFR)**

No matter what criterion is used, a good criterion should be able to tell apart and identify desired product for the ease of its users [5].

1.4 Non-Functional Requirements

The NFRs unlike functional requirements describe the way of assessing how the system operates instead of describing the functionality. They determine the quality of a piece of software. The terms non-functional requirements, constraints and quality attributes are often interchangeably used [6]. Some consider non-functional requirements a broader term that encapsulate various attributes. Figure 1 shows that non-functional requirements contain quality attributes, architectural attributes, domain attributes and organizational attributes as subsets[5], [7].

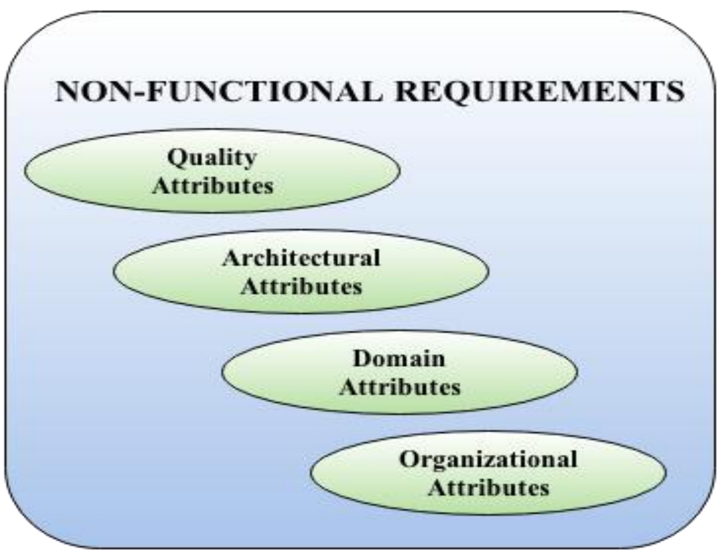

Figure 1: Non-Functional Requirements

1.5 Decision Support System (DSS)

As its name suggests, a DSS is a software tool that aids in decision making. It takes into consideration a number of factors and proposes the best possible solution to the user. DSS is in no way a substitute for human beings but help them in making complex decisions [8]. Such decisions are often critical and human beings are likely to skip important factors which should be otherwise taken into consideration. A Decision support system consists of the following elements:

- **A knowledge base or Database**
- **Model that specifies the criteria of evaluation**
- **A user interface that enables the users to interact with the DSS**

Above mentioned elements serve as the building blocks of any DSS architecture [8][9].

1.6 AHP

AHP stands for Analytic Hierarchy Process. It is a mathematical technique for examining and solving complex decision problems, which was invented by Thomas L. Saaty in 1900s. It is based on the principle of "divide and conquer". Consider the figure 2 for a better understanding:

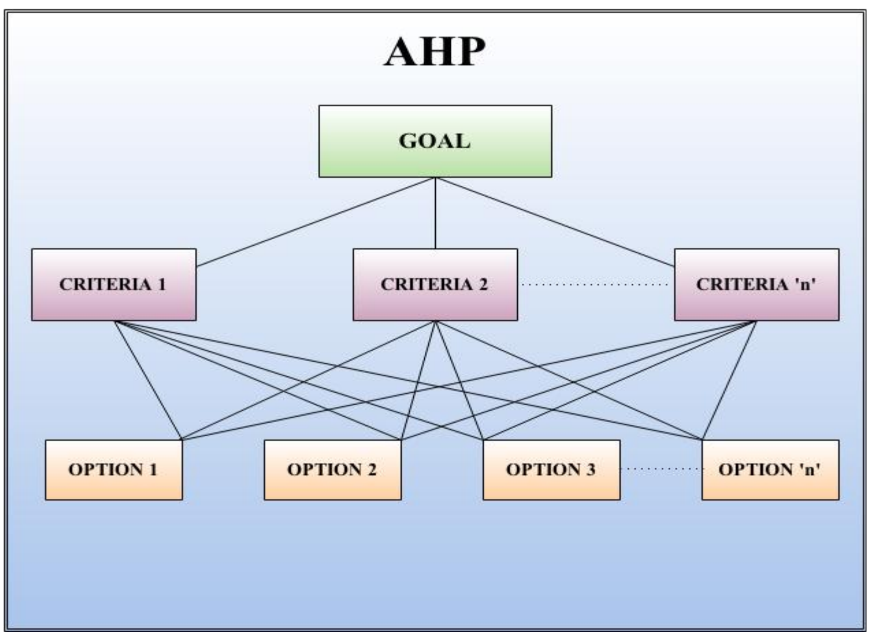

Figure 2: Analytic Hierarchy Process AHP

It is applied is situations where one option amongst a list of available alternatives is to be selected [10]. It uses a hierarchal approach for dividing a main problem into sub problems and then evaluates various different solutions that are present to solve the problem. The working of

AHP is described by example in Figure 2. Here the example of achieving a certain goal is presented. To achieve this goal the goal itself is divided into a number of criteria that will help in achieving it. Various different options are evaluated against these criteria and mapped to the overall goal [11]. The application of this technique is not bound to a single field. It can be applied in a number of areas including, education, business, healthcare and numerous other fields.

1.7 Rule-Based Reasoning

Rule-based reasoning refers to a kind of reasoning that makes use of statements that have "IF-THEN-ELSE" format. It is implemented using a rule-based system. In a rule-based system, inference mechanism tries to locate a rule that matches the desired pattern and then acts accordingly. Expert System is an example of such systems. The IF portion of the statement checks if the condition stands true, the THEN portion specifies taking a particular action, and the ELSE portion specifies an action which will be carried out in-case the IF stands FALSE. Consider the example:

IF points are greater than 100

THEN deduct 10

ELSE deduct 5.

Normally a rule-based system consists of [12]:

- **Rule Base (Knowledge base)**
- **Inference Engine**
- **Memory**
- **User Interface**

Basic Architecture is shown in Figure 3. User interacts with the system through a user interface. Based on that interaction and input, the inference engine takes an action. Inference engine first tries to match the input to the rule-base/knowledge base, if a match occurs; it looks for the action to be taken and then takes an action accordingly.

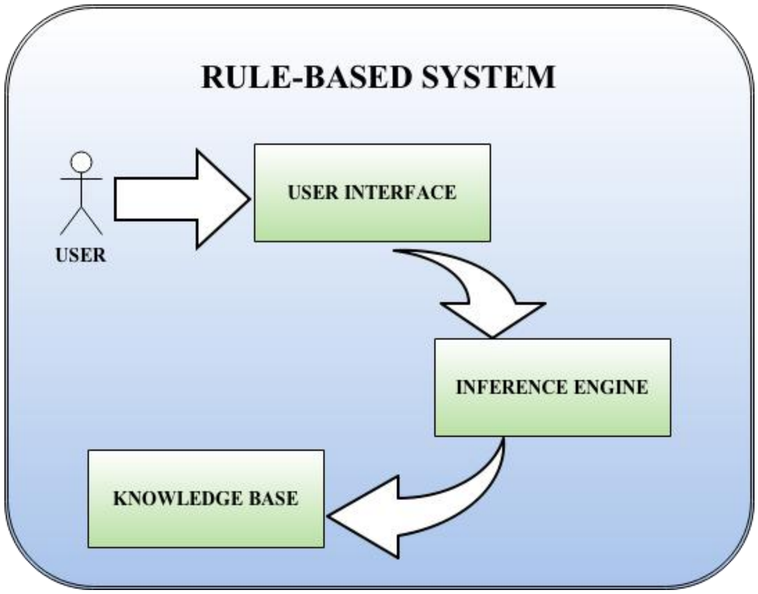

Figure 3: Rule-Based System Architecture

1.8 Background, Scope and Motivation

The Process of Software development is changing rapidly. With the evolution in technology and growth in software industry, user requirements have started to change exponentially faster. This has lead to the creation of a whole new approach that serves as a replacement for the traditional software development. This approach is implemented through the use of pre-existing software packages called COTS [5]. The success of such approach largely depends on the selection methodology that is being followed [13]. If chosen correctly, the use of such products can result in reduced cost, less development time and various other benefits [14], [15]. There are a number of available methods, each having its own advantages as well as limitations but unfortunately there is no method that fits in all situations and could be formalized [5], [16].

This research work is an effort to address various problems that are faced during the COTS evaluation and selection process. Questions like how and what should be done are answered in detail. A thorough approach is presented for the evaluation for products that is supported by a working prototype tool. Effort has been made to come up with an approach that has strengths of as much as possible number of methodologies and limitations of none.

1.9 Problem Statement

- The aim of this work is "to propose a comprehensive approach for systematic evaluation and selection COTS components." This can be broken down into two sub-tasks:
  - To propose an approach that evaluates COTS software based on both functional and non-functional requirements and assists the evaluators in choosing the optimal software product.
  - To propose an approach that is thorough enough to cover all aspects of COTS software selection and at the same time simple enough to be practically implementable.

1.10 Thesis Outline

The structure of this thesis is shown in figure 4. It has five chapters. Chapter 1 introduces its readers to various terminologies and concepts that are a part of this research work. Chapter 2 is literature review, which discusses different related methodologies and techniques. Chapter 3 and 4 are about the proposed approach and its implementation & results, respectively. Chapter 5 concludes the thesis.

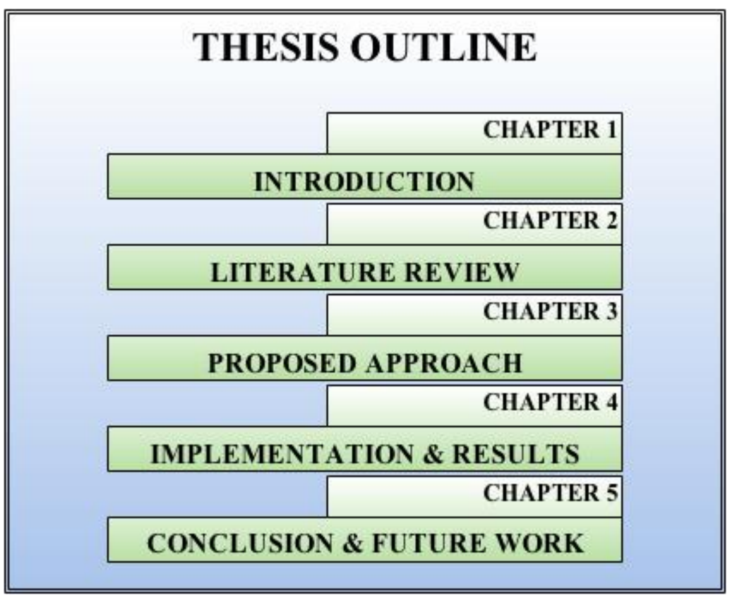

Figure 4: Thesis Outline

Chapter 2 Literature Review

This chapter discusses noteworthy contributions in literature relevant to evaluation and selection of COTS software. Section 2.1 discusses various approaches and section 2.2 contains a summary of best practices and common limitations found in most of the methodologies.

2.1 COTS Evaluation & Selection

A number of COTS evaluation methods have been proposed over the last couple of decades. Some propose evaluation criteria and some suggest, instruct and guide about making the evaluation efficient. In this chapter few of those methods have been discussed. Summary of each is given in the following sub-sections:

2.1.1 OTSO

OTSO (Off-The-Shelf-Option) [17] proposed by J. Kontio is a method for comparison of COTS products. It is one of the earliest methods and perhaps the most widespread method which serves as a basis for many other approaches present in literature. It is divided into six steps:

- Search
- Screen
- Evaluate
- Analyze
- Deploy
- Assess

It conducts cost-value analysis which serves as the criteria for evaluating COTS software. The analysis for each software product, under consideration, is carried out based on AHP model. It is often criticized for not being effective in cases where large numbers of products are to be evaluated since it uses AHP which uses pair-wise comparisons.

2.1.2 IusWare

IusWare (IUStitia softWARis) [18] proposed by M. Morisio and A. Tsoukias claims to be a COTS evaluation methodology that formally and thoroughly evaluates products. It is based on multi-criteria decision aid (also known as MCDM or MCDA) which is an area of operational research that deals with techniques to improve decision making capabilities. It works in two phases:

- Designing of model
- Implementation of the designed model.

The strength of this approach is that it considers quality attributes and its weakness is not taking into consideration, the user requirements.

2.1.3 CEP

The CEP (Comparative Evaluation Process) [19] proposed by Barbara Cavanaugh Phillips and Susan M. Polen is an approach established on spreadsheet model that helps evaluate the software components, particularly COTS. It introduced the use of credibility factor (CF) which takes into consideration the reliability of the data originator. Importance is given to the trustworthy sources, and the credibility factor in combination with the criteria determines the overall rating of a product. The process is shown in figure 5. It is based on five activities:

- Effort estimation
- Search & Screen
- Define Evaluation Criteria
- Evaluate
- Inspect results

It is praised for its use credibility factor but just like IusWare, it doesn’t focus on the requirements and presumes that they already exist.

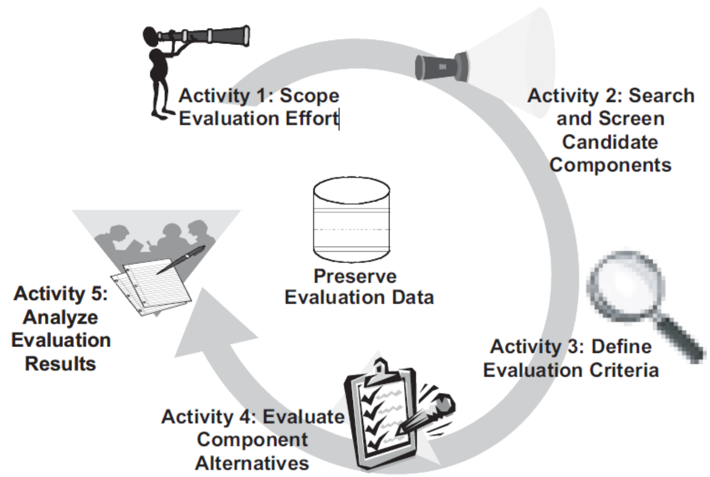

Figure 5: CEP (Comparative Evaluation Process) [19]

2.1.4 CRE

CRE (COTS based on Requirements Engineering) [20] is a requirement centered COTS selection approach by Carina Alves and Jaelson Castro. It selects by rejecting the unwanted candidates while working in iterations. It works in four phases:

- Identify
- Describe
- Evaluate
- Accept

The strength of this approach is that unlike few others it considers both functional and NFRs in evaluation process. This requirement driven approach is limited to a lower number of products under evaluation and fewer evaluation criteria. Cost value analysis is emphasized but guidelines to achieve such balance are not provided in this approach.

2.1.5 CAP

CAP (COTS Acquisition Process) [21] is a COTS selection method proposed by Michael Ochs et al. It claims to be a systematic and efficient method and suggests customization of not only the inspection criteria but the inspection process as well. The method has three basic elements:

- CAP-IC (Initialize)

- CAP-EC (Execute)
- CAP-RC (Reuse)

The best features of this method are its estimating the overall effort required and use of RC (Reuse Component) that will keep information helpful in future. But like few other methods it also assumes that the requirements are already available.

2.1.6 CARE

CARE (COTS-Aware Requirements Engineering) [22] is a process-oriented approach proposed by Chung Lawrence et al that targets the requirement stage in particular. It introduces the concept of requirement negotiation and argues the existence of requirements that may not be provided. The overview is given in the figure 6. It consists of five stages starting with objective definition, followed by search for COTS, classification of products, change settlement and finally selection of favorable candidate.

The strength of this approach is its ability to address the problem of mismatching issue. But it is also considered its weakness because not sufficient guidelines are provided to solve this problem.

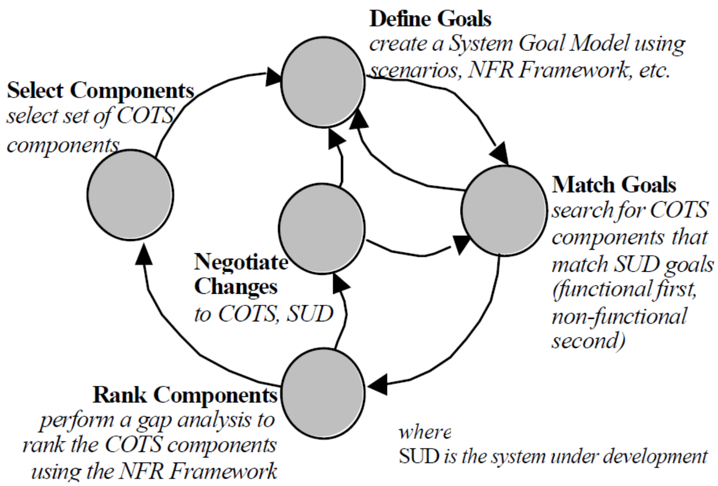

Figure 6: CARE (COTS-Aware Requirements Engineering) [22]

2.1.7 STACE

STACE (Social-Technical Approach to COTS Evaluation) [23] proposed by Douglas Kunda and Laurence Brooks highlights the importance of customer involvement in the evaluation of products. It argues the importance of non-technical issues that are often neglected in the selection process like social and institutional features. The basic framework is shown in figure 7.

The strength of this approach is taking into account the social and other non-technical issues but it requires an immense effort to practically implement it.

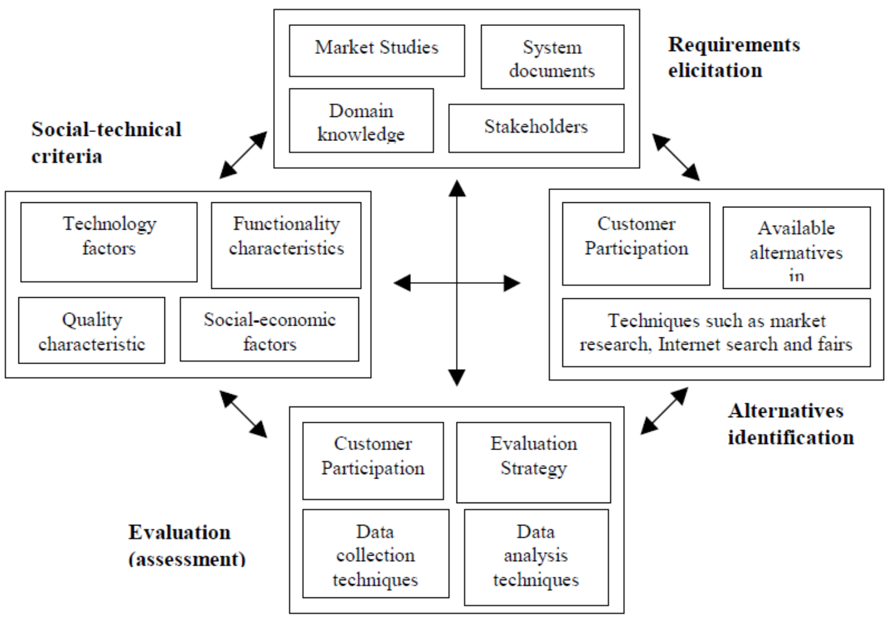

Figure 7: STACE [23]

2.1.8 PECA

It is an approach proposed by Santiago Comella-Dorda et al [24] that focuses on collecting the appropriate data necessary for the evaluation process instead of focusing on the evaluation. It gives detailed instruction about customization of the process to suit oneself. It consists of four steps:

- Plan
- Establish criteria

- Collect info
- Analyze data

Customizability can be considered its strength.

2.1.9 MIHOS

MiHOS (Mismatch Handling for COTS Selection) [25] proposed by Abdallah Mohamed et al targets the mismatches that are encountered during the selection process. It can be integrated with other approaches both during and after the selection and evaluation. It relies and focuses on functional requirements only. Also practically implementing this approach can prove to be a tedious job.

2.1.10 Storyboard

Storyboard approach [26] proposed by Sallie Gregor et al emphasizes on the use of use-cases and screenshots to gain complete knowledge of user requirements which will eventually lead to selection of most suitable COTS product. It does not provide an evaluation technique but suggests that clear requirements will result in less effort required for evaluation and future integration.

2.1.11 UnHOS

UnHOS (Uncertainty Handling in COTS Selection) [27] is a method for COTS evaluation proposed by Hamdy Ibrahim et al. It is a mixed approach based on AHP and BBN (Bayesian Belief Network). Probabilities are graphically shown in BBN models. UnHOS considers the quotient of uncertainty about the data provided for a COTS product in their selection process. AHP does all the evaluation whereas BNN is used to present the doubts associated. A tool supports the evaluation method.

Consideration of the uncertainty can be considered strength of this approach since it is a serious issue associated with COTS evaluation and selection.

2.1.12 W-Shaped Framework

Vinay et al proposed W-shaped framework [15] for component selection and product development process which provides guidelines for both the activities. This method addresses the issues faced during CBSD (COTS based software development). It considers both functional and NFRs for filtering out the candidates. The overall framework is shown in figure 8.

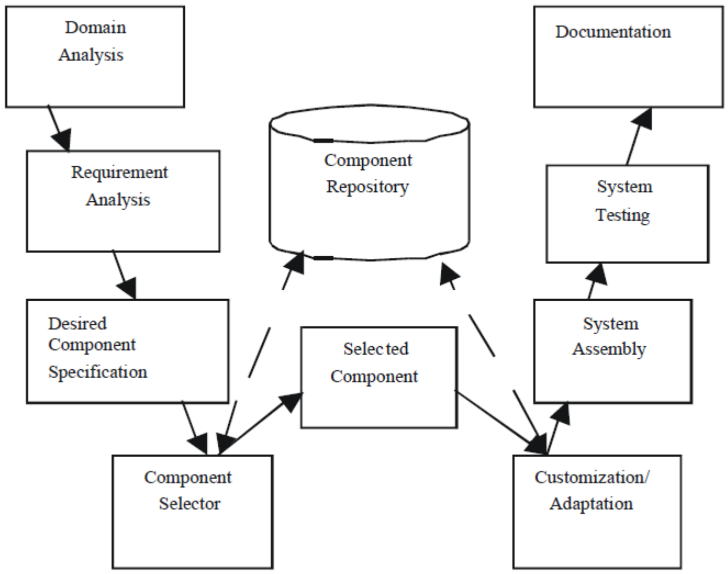

Figure 8: W-Shaped Framework [15]

2.1.13 IROTS

Idealize Recommendation Off-The-Shelf (IROTS) [14], proposed by M. Shakeel Faridi et al argues the lack of knowledge regarding COTS evaluation approaches in under development countries. It focuses on the use of ISO/IEC 25010 quality standard. It discusses issues associated with the CBSD, the categories of COTS product available in the market and proposes a nine stage process for product selection.

2.2 Discussion

Based on the critical analysis of the COTS evaluation & selection methods discussed in section 2.1 and other methods from the literature we can conclude that the general problems or limitations found in most of these approaches are:

2.2.1 Assuming User requirements Pre-Exist

Without well understood and properly obtained user requirements, it is not possible for the evaluators to produce quality results [28]. A number of COTS evaluation methods skip this very important phase [4]. Without focusing on this phase of the selection process, one increases the chances of the whole process to become a failure. A formal technique to collect and record requirements prior to the search of eligible candidates can guide the selectors to the path of success.

2.2.2 Having a Single Evaluation Criteria

Relying on a single evaluation criterion is another problem that can result in a disaster. One criterion might consider a particular aspect of a software system in depth but the evaluators would be overlooking a number of other factors, hence a quality product might get rejected and it would result into the selection of product that would lead to the failed selection process. Having multiple criteria gives evaluators the capability to better discriminate the software under inspection [5], [29].

2.2.3 Not Considering NFRs in the Evaluation

Not considering non-functional requirements while carrying out the assessment is considered a major shortfall of an approach [30]. NFRs are believed to be an important criterion of evaluation which plays an important part in efficiently differentiating the products [5], [31]. Evaluation based on consideration of cost or functional requirements will get the evaluator a product that seems to be a right choice but it may not essentially be the right and quality product. So, considering NFR's is a must in such evaluations.

2.2.4 Not Describing in Detail, What To Do?

One of the main reasons for the selection of COTS based on ad-hoc manner is the limited assistance provided by the approaches in literature [32]. Many approaches give general guidelines about how to conduct the selection, but do not describe in detail; what to do? How to do it? An implementable approach should be thorough enough and should provide guidelines about each activity conducted during the process.

2.2.5 Higher Complexity/Effort Level

One can say that there is a trade-off between effectiveness and complexity/Effort. Evaluators don't tend to practice approaches that have higher complexity level and require more effort to execute since such practices will be time-consuming. A supporting tool to assist in conducting the evaluation process reduces the overall effort and hence the time spent in it. But unfortunately only a couple of proposed approaches have a tool to support the process [33].

An approach that successfully deals with the above discussed issues can be considered to be practicing the best practices in COTS selection.

Summary

In this chapter we discussed several COTS evaluation & selection approaches. Strengths and weaknesses for each were discussed along with the way they work. In the end the common limitations found in most of the approaches were discussed.

Chapter 3 Proposed Approach

This chapter discusses the proposed approach in detail. Section 3.1 introduces the approach and section 3.2 contains its detailed discussion.

3.1 TAES-COTS

Thorough approach for evaluation and selection of COTS products (TAES-COTS) is an approach that evaluates COTS products based on progressive filtration. It is a five step approach that evaluates products against different criteria at three different stages. Figure 9 shows the overall process. The process is divided into five steps that are briefly described in section 3.1.1 and discussed in detail in section 3.2.

3.1.1 Main Steps

- **STEP 1:** First step of this approach is the preparatory step. Activities like requirement gathering and roles definition are performed in this approach.
- **STEP 2:** Second step is searching for the potential candidates to evaluate. Activities related to search of COTS products that seem to meet the criteria are performed here.
- **STEP 3:** Third step is the first evaluation that screens out the COTS products based on functional requirements and cost factor.
- **STEP 4:** Fourth step is second evaluation that is the detailed COTS evaluation based on non functional requirements. The products that meet the criteria in step 3 are further evaluated here. This is the most important step in the evaluation process.
- **STEP 5:** The last Step is the final and optional evaluation that results in the selection of most suitable candidate. After this evaluation the process is documented and then terminated.

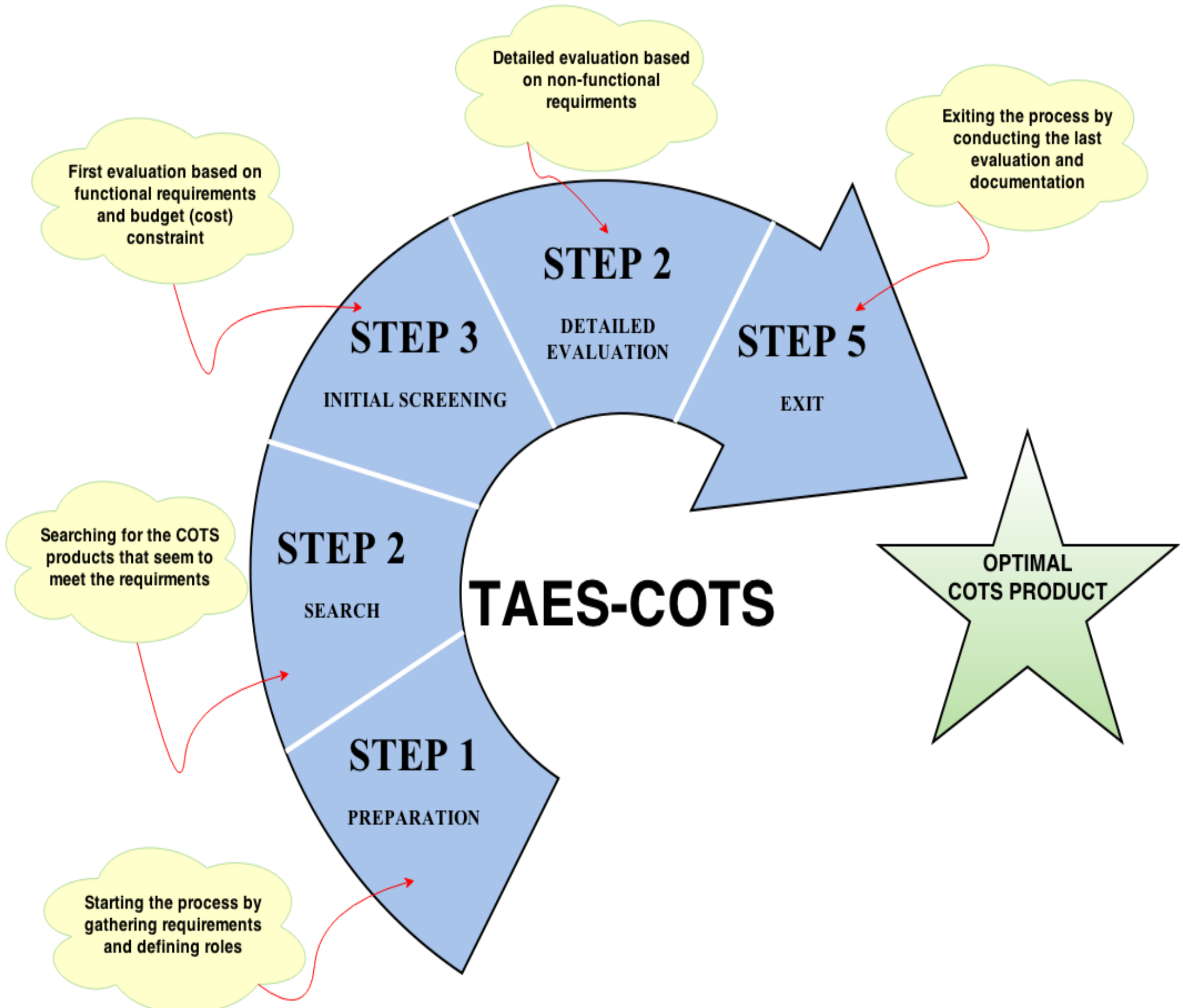

Figure 9: TAES-COTS Approach

3.1.2 Evaluation Criteria

The complete set of criteria and a product's journey throughout these criteria is shown in figure 10. The products are first evaluated in step 3 based on functional requirements and budget constraint/cost factor. The products meeting the criteria are forwarded to step 4 for further evaluation based on the set of non-functional requirements and the rest are rejected. After these two evaluations, if a single product with significantly higher score then others is found, it is selected. Otherwise, the top few products are again evaluated based on their vendor information.

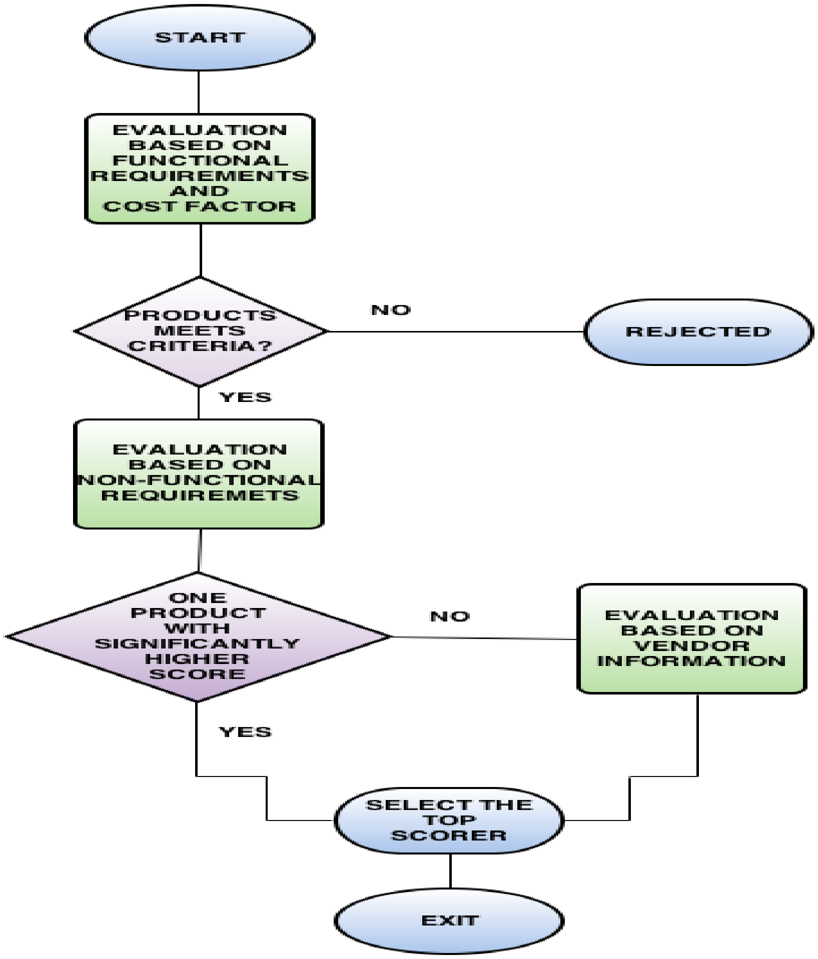

Figure 10: Complete Criteria

3.2 TAES-COTS in Detail

This section discusses all the five steps of TAES-COTS in detail.

3.2.1 STEP 1

The first step of this approach is the preparation step. In this step first the roles and responsibilities are defined. And then user requirements are gathered.

3.2.1.1 Roles and Responsibilities

The team that has been assigned the task to evaluate and select COTS products will have more than one member in it. It is important that each of them should be aware of his/her responsibilities. Team needs to manage its manpower efficiently so that the selection is conducted well in time and efficiently. The responsibilities of a single member will differ based on the size of the team. In larger teams one member might only perform a single duty, but in considerably smaller teams, the responsibilities will increase accordingly. M. Morisio et al [34] and [35] discuss various roles and responsibilities. For our approach, following key responsibilities should be divided amongst the team members before starting a formal evaluation process:

- **Requirement gathering (Functional):** Requirement engineering is the process of gathering and recording the user requirements. The person(s) responsible for gathering the requirements should have knowledge of techniques associated with this process. Requirements regarding the desired functionality provided by the product are to be collected from the user by the person responsible.
- **Searching the products:** The search for the COTS products is another responsibility. The person responsible for this activity should have knowledge and access to the places where products are likely to be found.
- **Collecting evaluation data (Non-Functional):** Gathering input data that will be used in evaluating the products is perhaps amongst the most important and lengthy tasks. The person responsible should gather data for each product under evaluation very carefully. If more than one person is assigned the duty, they should work in pairs for better results.
- **Conducting the evaluation:** Preferably, more than one person should be assigned the task to perform this duty. Since this activity will decide the final output of the process, ideally an experienced, knowledgeable and cautious person(s) should be assigned this duty.
- **Documenting the process:** A person should be assigned the task to record the overall process. Such information might prove helpful in future selections.
- **Managing overall process:** A person should be responsible to manage the overall process and keep a check on each activity and also check progress from time to time. And divide the work amongst the individuals as well. Usually a project manager performs such duties.
- **Checking quality of work:** Additionally, a person should take the responsibility of keeping a check on the quality of work being conducted during the process.

Other roles and responsibilities can be formed according to needs. The interaction amongst the individuals involved is a must.

3.2.1.2 Requirement Gathering

Many approaches present in the literature do not discuss this very important activity of COTS software selection called requirement gathering. Before the evaluators can evaluate the products, they should know exactly what they should be evaluating. Only correct requirements can lead the process to the selection of correct product. Combination of questionnaires, use-cases and storyboards is proposed for gathering functional requirements. Using multiple techniques in combination will help gather complete and accurate requirements besides that; different techniques will overcome the limitations of each other. Example of each is shown in figure 11. Sallie Gregor et al [26] discussed benefits of using storyboard and use-case together in detail. The brief discussion about each is as under:

- **Questionnaire:** It is one of the requirements gathering techniques that comprises of a series of questions. It is a cheap technique and requires considerably less effort to collect information. But the information gathered via a questionnaire is limited to the questions asked in it. It should particularly be used to clarify vague information provided by the user.
- **Use-case:** A use-case graphically shows the functionalities provided by system in general from a user's perspective. A use-case may not describe in detail how the system works but will definitely describe the overall features present in a system.
- **Storyboard:** Storyboard is an efficient way of collecting user requirements. It shows how the system will work in detail with the focus on GUI. Mockups, drawing or screen captures from similar software are the contents of a storyboard.

Showing the storyboards and use-cases to the user and taking feedback from them on the spot will help a lot in getting the complete requirements. Once the requirements are gathered, it is necessary that the user signatures be taken on each of these documents to conclude the requirement gathering. Note that the focus of this requirement gathering stage will be to capture functional requirements and information regarding budget constraint for the first evaluation performed in step 3.

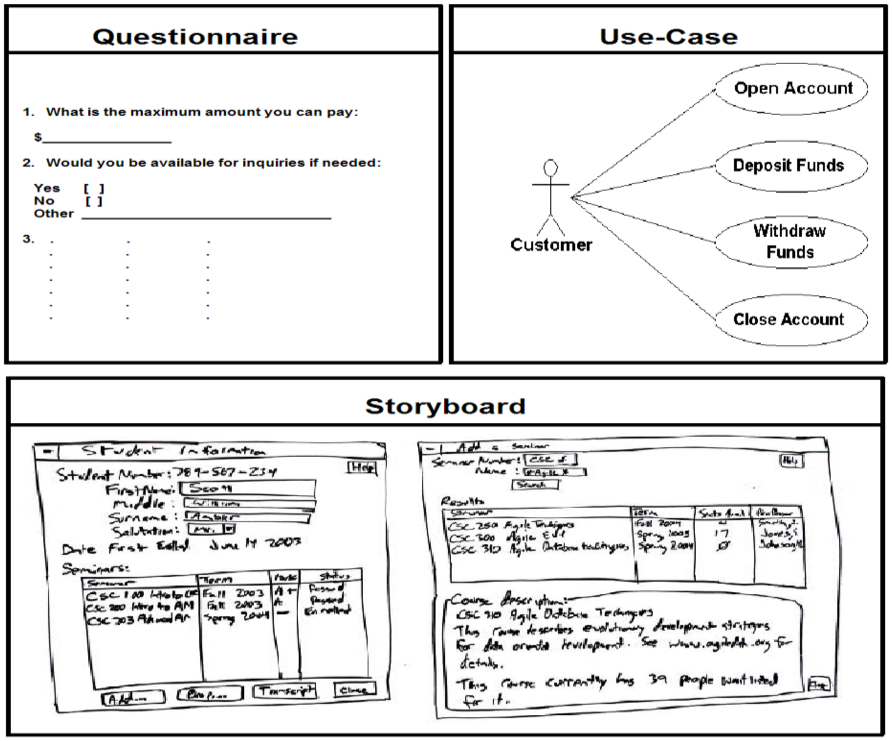

Figure 11: Questionnaire, Use-case and Storyboard Example

3.2.2 STEP 2

The next step is to search for potential candidates. The main objective of this step should be gathering information about as much as possible COTS products that seem to meet the criteria instead of going into the details of whether or not they meet the criteria and at the same time keeping the search specific [36]. Going into details at this stage of the process will not only result in more time required to complete this step but also result in missing out the potential candidates that might have been the required product. Search might be internal or external [19]. The internal and external search areas are:

- **Internal Search Areas:** If the evaluation team works for an organization that has a repository of COTS products, or database containing information regarding COTS products, it is preferable to first look there for software. If not the previous case, another area to look for product's information can be the documents containing information about the evaluations performed in the past. Not only will it save time, but also information regarding non-functional requirements needed for evaluation in step 3 will be acquired without any effort. The information collected internally will be more reliable since it will be information based on in-house use of the products.

- **External search Areas:** Looking online for potential candidates is always an option. Using search engines might help find a number of candidates easily. Using keywords in a search engine will suggest a number of packages in no time. Advantages of looking up for packages online is that not only a greater number of products can be found but also the products that are newest in market and up-to-date in technology can be spotted pretty quickly. Besides looking online, organizations that develop software can be contacted and asked to provide information about their products. Also other organizations that have friendly relations with the evaluating organization might be requested for such information if they keep any.

3.2.3 STEP 3

Once the information about probable candidates has been collected, it is now time to evaluate them. This step serves as initial screening of the products; they will be evaluated for the very first time here. This step considers functional requirements & cost factor as evaluation criteria and this is where the requirements gathered in step-1 will come into play.

Suppose we have total ‘n’ number of candidates for inspection, if ‘X’ represents the set of all the candidates and ‘x’ represents a single candidate, then:

$$\boldsymbol{X = x1, x2, x3, \dots, xn} \qquad \dots \text{ Eq-3.1}$$

Further suppose that we have total “t” number of functional requirements, if ‘Y’ is a set of functional requirements that a candidate must possess, ‘y’ represents a single functional requirement and ‘Z’ is the Budget restriction (i.e. upper limit for the cost of a COTS product) then:

$$\boldsymbol{Y = y1, y2, y3, \dots, yt} \qquad \dots \text{ Eq-3.2}$$

Now each candidate ‘x’ will be examined for Y and Z (i.e. both functional requirements and budget constraint) and only those candidates will be forwarded for further evaluation in step 4 which fully meet the criteria here.

After stage 3, we will have ‘S’ candidates, such that:

$$\boldsymbol{S} = \boldsymbol{X} - \boldsymbol{R} \qquad \ldots \text{ Eq-3.3}$$

Where, ‘R’ is the set of rejected candidates and ‘S’ represents the set of selected candidates (candidates that meet the Y+Z criteria). Figure 12 graphically shows the above discussion.

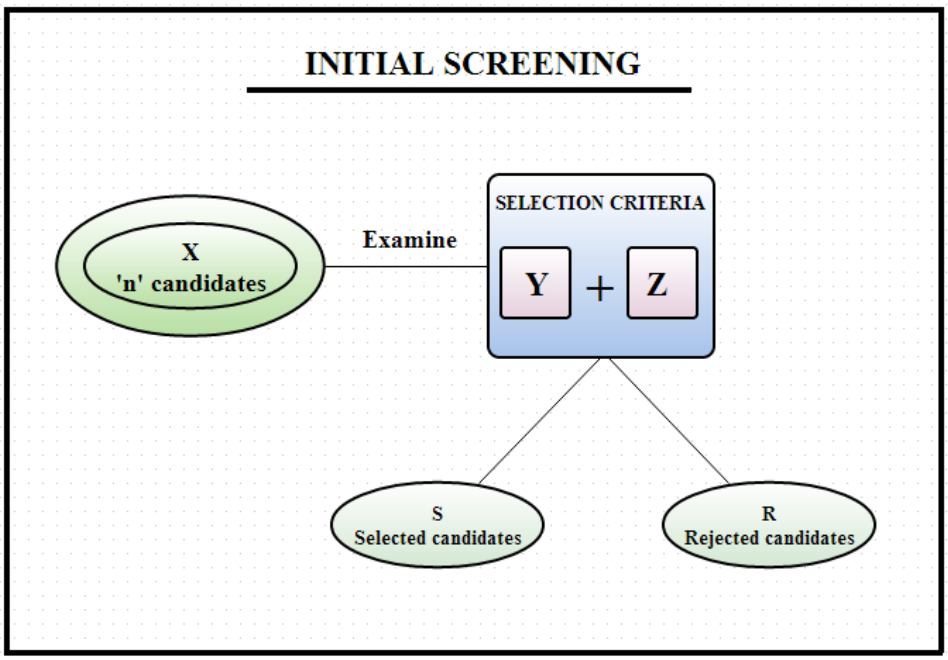

Figure 12: Initial Screening

This evaluation serves two purposes:

- It eliminates the irrelevant products that can’t be the suitable choice.
- It narrows down the list of products that have to be evaluated in detail in the step-4.

Since only two factors would be considered in evaluation, and only presence or absence of a requirement will be checked in each product without having to have complex mathematical calculations; it can easily be done by making a table with checklist. A template has been provided in Appendix-A (Table A-1). One key thing to note here is that the functional requirements have been divided into two lists: mandatory and optional requirements. At the time of requirement gathering the users must be enquired about the list of functional requirements that they consider essential for their desired software and the list of those that are optional. Now, while evaluating the software, the evaluators will reject the products that don't meet the mandatory requirements, even if they meet the budget constraint and all the optional requirements. Only those products will be selected for further evaluations which meet both mandatory functional requirements and budget constraint. But the products which meet both mandatory and optional requirements along with the cost factor will be given preference in the final selection decision after all the evaluations have been conducted. Use of a special flag or tag is suggested for each candidate meeting the criteria. Table-1 shows the flags:

Table 1: Priority Flags

<table>
<tr><th></th><th>Products that meet:
Mandatory Functional Requirements
Optional Functional Requirements
Budget Constraint</th><th>Products that meet:
Mandatory Functional Requirements
Budget Constraint</th></tr>
<tr><td>Flags/Tag</td><td>FPS
(First Priority in selection)</td><td>SPS
(Second Priority in selection)</td></tr>
</table>

These flags will play a key role in final selection if multiple products with equal scores are encountered during the detailed evaluation. In such cases, preference will be given to products carrying FPS flag.

3.2.4 STEP 4

This step is the main focus of this research work. Here products will be evaluated in detail based on non-functional requirements. Evaluation based on functional requirement is not a difficult task to perform, anyone with sufficient knowledge of software can perform such evaluation by checking the presence of a particular functionality. But consider a scenario where multiple products meet the functional criteria. How should a product be selected amongst them? In such uncertainty, it is essential to perform an evaluation that selects quality software. This step addresses the same issue.

The underlying logical model for this evaluation is AHP and the quality model being considered is proposed by Alexandre Alvaro et al [37], [38].

3.2.4.1 Quality Model

As discussed in section 1.4, non-functional requirement is a broad concept that encompasses quality attributes, architectural attributes, domain and organizational attributes. The problem with evaluating software based on such characteristics is their qualitative nature. It is important to find a way and convert these qualitative characteristics into quantitative form. Alexandre alvaro et al [37] [38] proposed a quality model that divides the basic non-functional requirements into sub-characteristics and further into sub-sub-characteristics or attributes such that those attributes are directly measurable. The model has been altered to suit our approach. It is not practical to use all of the attributes proposed by them, so, after detailed literature survey and discussion with various professionals, 18 attributes have been selected for use in our approach. The reason for choosing these attributes was to eliminate the irrelevant or "difficult to measure" attributes and make it possible for the evaluators to collect data for these attributes easily for every product under inspection. But at the same time the aim was to retain all those attributes that comprehensively evaluate the products considering all aspects of software quality. Each of these 18 attributes belongs to one of the six basic characteristics, which are:

i. Functionality
ii. Reliability
iii. Efficiency
iv. Usability
v. Maintainability
vi. Portability

The attributes, what they mean in our approach and their parent characteristics are shown in Table 2.

Table 2: Attributes and Their Parent Characteristics

S-No	Attributes	Definition	Parent Characteristic
1.	Data Encryption	The data encryption mechanism for protecting data	Functionality
2.	Auditability	Mechanism for keeping track of users who access the data or system	Functionality
3.	Error Handling	Mechanism for handling the errors(e.g. Exception handling)	Reliability
4.	Response time	The time between getting a request and responding to that request	Efficiency
5.	Documentation available	Documentation, Demos, API's, Tutorials which assist is understanding the COTS product	Usability
6.	Testability	Test suits, Info about the errors found, Confirmation that formal testing was done for that component	Maintainability
7.	Memory utilization	The memory needed by the component	Efficiency
8.	Disk utilization	Disk space needed by the component	Efficiency
9.	Modifiability	The level of modification that can be done to a component till it can function properly	Maintainability
10.	Operability	Complexity and effort to operate a component	Usability
11.	Customizability	Number of customizable attributes that a component offers	Maintainability
12.	Deployability	Effort to deploy a component in a particular environment	Portability
13.	Standardization	If a component conforms to standards or laws	Functionality
14.	Certification	If a component is certified by any organization	Functionality
15.	Configuration capacity	Effort to transfer a component to other environments	Portability
16.	Dependability	If a component is dependent on any other component to provide its services	Functionality
17.	Learnability	Time and effort to learn to use/configure the component services	Usability
18.	Configurability	Effort and time to configure the component correctly	Usability

3.2.4.2 AHP and Quality Model

As discussed earlier the proposed evaluation is based on AHP. AHP as described in section 1.6 divides the goal into sub-goals and if required, sub-sub-goals. In our approach the goal is to "Select the optimal COTS product" and to achieve this goal four priority classes have been used. Each of these classes comprises different attributes discussed in previous section. The classes along with the attributes they have are shown in table-3, which is as under:

Table 3: Attributes Class Division

CLASSES	ATTRIBUTES
CLASS-A	Data Encryption
	Auditability
	Error Handling
	Response time
	Documentation available
	Testability
CLASS-B	Memory utilization
	Disk utilization
	Modifiability
	Operability
	Customizability
CLASS-C	Deployability
	Standardization
	Certification
	Configuration capacity
CLASS-D	Dependability
	Learnability
	Configurability

Not all aspects of a single characteristic are equally important, some are preferred over others. Their distribution in classes help in prioritizing these attributes. The class-A is the top priority class, followed by class-B. Class-C is third in priority and class-D the last. The AHP model here gets the form shown in Figure 13. The value of each attribute will determine the score of class that it belongs to, and the classes together will determine the overall suitability of a COTS product. Unlike traditional AHP comparisons, our approach uses a modified version and relies on rule-based reasoning for comparisons.

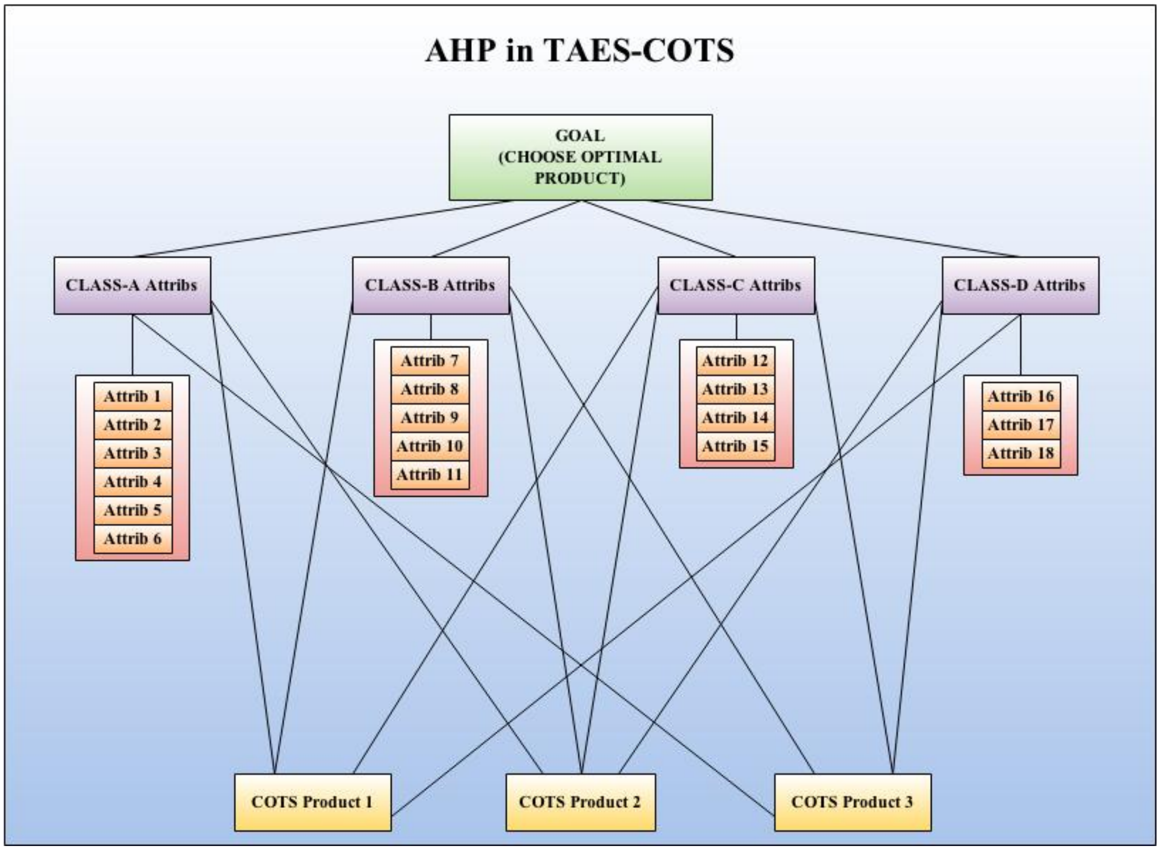

Figure 13: AHP in TAES-COTS

3.2.4.3 Attributes, Class Values and Rule-Base

The values of attributes determine score of classes and the classes eventually show the overall suitability of a product under evaluation. The range of values that can be entered for each attribute is between 0 and 1. Whereas the score of a class is calculated by taking average of the values of attributes. The equation for calculation of class's score is as under:

$$\text{ClassScore} = \frac{1}{n} * \sum_{i=1}^{n} \text{Atrrib}_i \quad \ldots \text{ Eq-3.4}$$

Where $Attrib_i$ is the value of a single attribute and "n" is the total number of attributes present in a class. The metrics used to measure the attributes are:

- Presence: P(Present)/A(Absent)
- Level: L(Low)/H(High)

To determine the value of an attribute, the evaluators should check the presence or level of an attribute. The metrics, value ranges and desirable values for each attribute are given in table-4.

Table 4: Attribute's Metric, Value, and Desirability

S-No	Attributes	Metric	Values	Desirability
1.	Data Encryption	(P/A)	(0 to 1) (1.0 for P/0.0 for A)	Presence is desirable
2.	Auditability	(P/A)	(0 to 1) (1.0 for P/0.0 for A)	Presence is desirable
3.	Error Handling	(P/A)	(0 to 1) (1.0 for P/0.0 for A)	Presence is desirable
4.	Response time	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
5.	Documentation available	(P/A)	(0 to 1) (1.0 for P/0.0 for A)	Presence is desirable
6.	Testability	(P/A)	(0 to 1) (1.0 for P/0.0 for A)	Presence is desirable
7.	Memory utilization	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
8.	Disk utilization	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
9.	Modifiability	(L/H)	(0 to 1) (0.0 for L/1.0 for H)	High level desirable
10.	Operability	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
11.	Customizability	(L/H)	(0 to 1) (0.0 for L/1.0 for H)	High level desirable
12.	Deployability	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
13.	Standardization	(P/A)	(0 to 1) (1.0 for P/0.0 for A)	Presence is desirable
14.	Certification	(P/A)	(0 to 1) (1.0 for P/0.0 for A)	Presence is desirable
15.	Configuration capacity	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
16.	Dependability	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
17.	Learnability	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable
18.	Configurability	(L/H)	(0 to 1) (1.0 for L/0.0 for H)	Low level desirable

Based on the score of a class, its value is determined as under:

- If (ClassScore >= 0.0 & ClassScore <= 0.4) then ClassValue = L(Low)
- If (ClassScore > 0.4 & ClassScore <= 0.7) then ClassValue = M(Medium)
- If (ClassScore > 0.7 & ClassScore <= 1.0) then ClassValue = H(High)

And the values of these classes determine the overall suitability of a COTS product. Values of these classes are taken as input and matched against the 81 rules. The 81 rules are given by the formula 3^4, where 3 is the total number of values a class can have (L/M/H) and 4 is the total number of classes (A, B, C and D). The input can be any of the three values (i.e. L/M/H) and output will be any of the following five values describing the COTS product appropriateness:

- L (Low)
- LM (Low-Medium)
- M (Medium)
- MH (Medium-High)
- H (High)

A single block of the rule base is as under:

IF		
CLASS-A VALUE	=	**LOW**
AND		
CLASS-B VALUE	=	**LOW**
AND		
CLASS-C VALUE	=	**LOW**
AND		
CLASS-D VALUE	=	**LOW**
THEN		
COTS SUITABILITY	=	**LOW**

Table 5 shows the overall suitability of a COTS product considering different values for each class based on the 81 rules.

Table 5: The Rules

S-No	Class-A Value	Class-B Value	Class-C Value	Class-D Value	COTS Suitability
1	A (Low)	B (Low)	C (Low)	D (Low)	L(Low)
2	A (Low)	B (Low)	C (Low)	D (medium)	L(Low)
3	A (Low)	B (Low)	C (Low)	D (High)	L(Low)
4	A (Low)	B (Low)	C (medium)	D (Low)	L(Low)
5	A (Low)	B (Low)	C (medium)	D (medium)	L(Low)
6	A (Low)	B (Low)	C (medium)	D (High)	L(Low)
7	A (Low)	B (Low)	C (High)	D (Low)	L(Low)
8	A (Low)	B (Low)	C (High)	D (medium)	L(Low)
9	A (Low)	B (Low)	C (High)	D (High)	L(Low)
10	A (Low)	B (medium)	C (Low)	D (Low)	L(Low)
11	A (Low)	B (medium)	C (Low)	D (medium)	L(Low)
12	A (Low)	B (medium)	C (Low)	D (High)	L(Low)
13	A (Low)	B (medium)	C (medium)	D (Low)	L(Low)
14	A (Low)	B (medium)	C (medium)	D (medium)	L(Low)
15	A (Low)	B (medium)	C (medium)	D (High)	L(Low)
16	A (Low)	B (medium)	C (High)	D (Low)	L(Low)
17	A (Low)	B (medium)	C (High)	D (medium)	L(Low)
18	A (Low)	B (medium)	C (High)	D (High)	LM(Low-Medium)
19	A (Low)	B (High)	C (Low)	D (Low)	LM(Low-Medium)
20	A (Low)	B (High)	C (Low)	D (medium)	LM(Low-Medium)
21	A (Low)	B (High)	C (Low)	D (High)	LM(Low-Medium)
22	A (Low)	B (High)	C (medium)	D (Low)	LM(Low-Medium)
23	A (Low)	B (High)	C (medium)	D (medium)	LM(Low-Medium)
24	A (Low)	B (High)	C (medium)	D (High)	LM(Low-Medium)
25	A (Low)	B (High)	C (High)	D (Low)	LM(Low-Medium)
26	A (Low)	B (High)	C (High)	D (medium)	LM(Low-Medium)
27	A (Low)	B (High)	C (High)	D (High)	LM(Low-Medium)

28	A (medium)	B (Low)	C (Low)	D (Low)	LM(Low-Medium)
29	A (medium)	B (Low)	C (Low)	D (medium)	LM(Low-Medium)
30	A (medium)	B (Low)	C (Low)	D (High)	LM(Low-Medium)
31	A (medium)	B (Low)	C (medium)	D (Low)	LM(Low-Medium)
32	A (medium)	B (Low)	C (medium)	D (medium)	LM(Low-Medium)
33	A (medium)	B (Low)	C (medium)	D (High)	LM(Low-Medium)
34	A (medium)	B (Low)	C (High)	D (Low)	LM(Low-Medium)
35	A (medium)	B (Low)	C (High)	D (medium)	LM(Low-Medium)
36	A (medium)	B (Low)	C (High)	D (High)	LM(Low-Medium)
37	A (medium)	B (medium)	C (Low)	D (Low)	M(medium)
38	A (medium)	B (medium)	C (Low)	D (medium)	M(medium)
39	A (medium)	B (medium)	C (Low)	D (High)	M(medium)
40	A (medium)	B (medium)	C (medium)	D (Low)	M(medium)
41	A (medium)	B (medium)	C (medium)	D (medium)	M(medium)
42	A (medium)	B (medium)	C (medium)	D (High)	M(medium)
43	A (medium)	B (medium)	C (High)	D (Low)	M(medium)
44	A (medium)	B (medium)	C (High)	D (medium)	M(medium)
45	A (medium)	B (medium)	C (High)	D (High)	M(medium)
46	A (medium)	B (High)	C (Low)	D (Low)	M(medium)
47	A (medium)	B (High)	C (Low)	D (medium)	M(medium)
48	A (medium)	B (High)	C (Low)	D (High)	M(medium)
49	A (medium)	B (High)	C (medium)	D (Low)	M(medium)
50	A (medium)	B (High)	C (medium)	D (medium)	M(medium)
51	A (medium)	B (High)	C (medium)	D (High)	M(medium)
52	A (medium)	B (High)	C (High)	D (Low)	MH(Medium-High)
53	A (medium)	B (High)	C (High)	D (medium)	MH(Medium-High)
54	A (medium)	B (High)	C (High)	D (High)	MH(Medium-High)
55	A (High)	B (Low)	C (Low)	D (Low)	MH(Medium-High)
56	A (High)	B (Low)	C (Low)	D (medium)	MH(Medium-High)
57	A (High)	B (Low)	C (Low)	D (High)	MH(Medium-High)

58	A (High)	B (Low)	C (medium)	D (Low)	MH(Medium-High)
59	A (High)	B (Low)	C (medium)	D (medium)	MH(Medium-High)
60	A (High)	B (Low)	C (medium)	D (High)	MH(Medium-High)
61	A (High)	B (Low)	C (High)	D (Low)	MH(Medium-High)
62	A (High)	B (Low)	C (High)	D (medium)	MH(Medium-High)
63	A (High)	B (Low)	C (High)	D (High)	MH(Medium-High)
64	A (High)	B (medium)	C (Low)	D (Low)	MH(Medium-High)
65	A (High)	B (medium)	C (Low)	D (medium)	MH(Medium-High)
66	A (High)	B (medium)	C (Low)	D (High)	MH(Medium-High)
67	A (High)	B (medium)	C (medium)	D (Low)	H(High)
68	A (High)	B (medium)	C (medium)	D (medium)	H(High)
69	A (High)	B (medium)	C (medium)	D (High)	H(High)
70	A (High)	B (medium)	C (High)	D (Low)	H(High)
71	A (High)	B (medium)	C (High)	D (medium)	H(High)
72	A (High)	B (medium)	C (High)	D (High)	H(High)
73	A (High)	B (High)	C (Low)	D (Low)	H(High)
74	A (High)	B (High)	C (Low)	D (medium)	H(High)
75	A (High)	B (High)	C (Low)	D (High)	H(High)
76	A (High)	B (High)	C (medium)	D (Low)	H(High)
77	A (High)	B (High)	C (medium)	D (medium)	H(High)
78	A (High)	B (High)	C (medium)	D (High)	H(High)
79	A (High)	B (High)	C (High)	D (Low)	H(High)
80	A (High)	B (High)	C (High)	D (medium)	H(High)
81	A (High)	B (High)	C (High)	D (High)	H(High)

3.2.5 STEP 5

This is the last stage of propose approach. Ideally, there will be a single optimal COTS product that will be a leading scorer after evaluation conducted in previous step. In such cases, the selection process terminates and the whole process is documented for future references. But in situations where multiple products have equal scores and they are shown to be the right choices, they need to be further evaluated. Here, such products are evaluated based on their vendor's information.

3.2.5.1 Evaluation Based on Vendor Information

The criteria basically consider the vendor organization's repute and maturity in the market. A number of characteristics might be considered for evaluation:

- **Time since launch:** The time of vendor organization in the market.
- **Products available in market:** Total number of software products the organization has produced
- **Well known products:** Number of well-known or successful products
- **User's comments (word on the street):** What do the clients of the organization say about them
- **Organization size:** Number of employees or number of locations that the organization operates at.
- **Support availability:** Does the organization provide any technical support to its customers.
- **Past experiences:** Any purchases or interactions with the vendor organization in the past. If any, how was the experience?

Other characteristics might be added to suit one's requirements. Since there will be a small number of products under evaluation in this stage, the evaluation can easily be done using a similar approach proposed in step-3. A template has been provided in Appendix-A (Table A-2). For each entry in the table, the score of 1 shall be given to the product(s) with desirable value and zero to the rest. In the end, after calculating the total score for each product, the product with highest score will be the most desirable choice.

3.2.5.2 Documenting the process

[19] Discusses the importance of documentation. All details shall be documented after the termination of the process. The data about all the products that were considered, the notes about carrying out a particular step in some specific manner, the problems faced in the evaluation and every other minor detail. It will prove to be helpful in future evaluations, be it the same team

carrying out another selection process or a team with new members. The documentation style and format is out of the scope of this research work.

The main activities of this approach along with the order in which they are performed can be seen in figure-14.

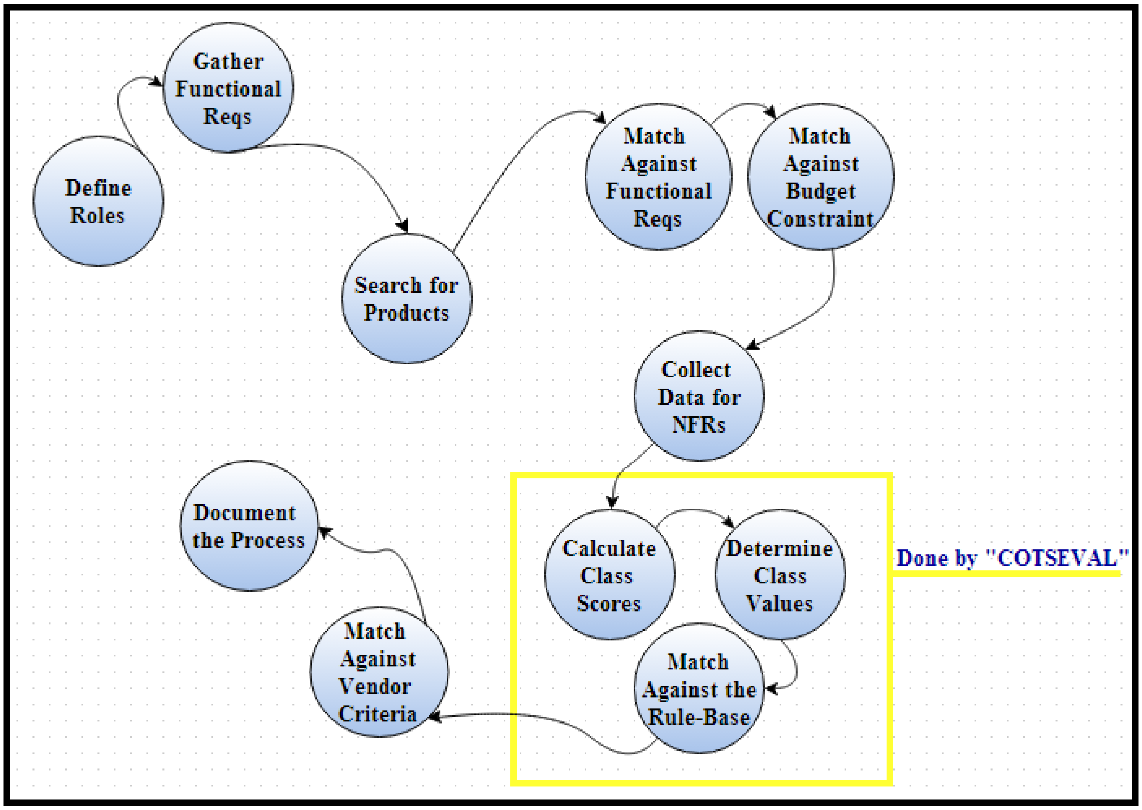

Figure 14: TAES-COTS Main Activities

Summary

TAES-COTS is an approach that evaluates the COTS products thoroughly and rigorously. In this chapter the proposed approach i.e. TAES-COTS was discussed. Each of its five steps was discussed in detail and guidelines were given to efficiently implement each.

Chapter 4 Implementation and Validation

This chapter discusses the software-tool "COTSEVAL", developed to assist the evaluators in step-4 evaluation. It also contains analysis of the results and comparison of TAES-COTS with other approaches. Section 4.1 contains an introduction of the "COTSEVAL" and discusses its user interface. Section 4.2 discusses the validation of proposed approach.

4.1 COTSEVAL

4.1.1 Introduction

"COTSEVAL: A Tool for Evaluating COTS Products" is a prototype software that evaluates COTS products based on the criteria defined in section 3.2.4. It helps in carrying out the detailed evaluation of COTS products in the fourth step of the proposed approach. The evaluation done in steps 3 and 5 can easily be done using the templates provided in Appendix-A. But carrying out the detailed evaluation can prove to be a lengthy and tedious job. So to assist in evaluation and reduce the evaluation time from days to hours and minutes, this prototype tool has been developed. The functionalities provided by this tool are shown using a use-case diagram in figure-15.

Using this tool, a user can perform three basic tasks, which are: User management, COTS product management and Comparison management. In user management, a user of the system can add new users, modify information about existing users and delete a user as well. In COTS management area, a user can add information about a Product, modify information and view information about a particular product. In Comparison area, a user can perform comparisons, save them and view information about past comparisons.

COTSEVAL not only serves as a support tool for performing detailed evaluation but also serves as a data repository for COTS products that may or may not be a part of any comparison activity. Such information about a product can prove to be useful in future comparisons and will save an ample amount of time for the evaluators.

The tool has been developed in C# using Microsoft Visual Studio, and database has been created using Microsoft SQL Server.

The functionalities and User interface of the tool are discussed in detail in the next section.

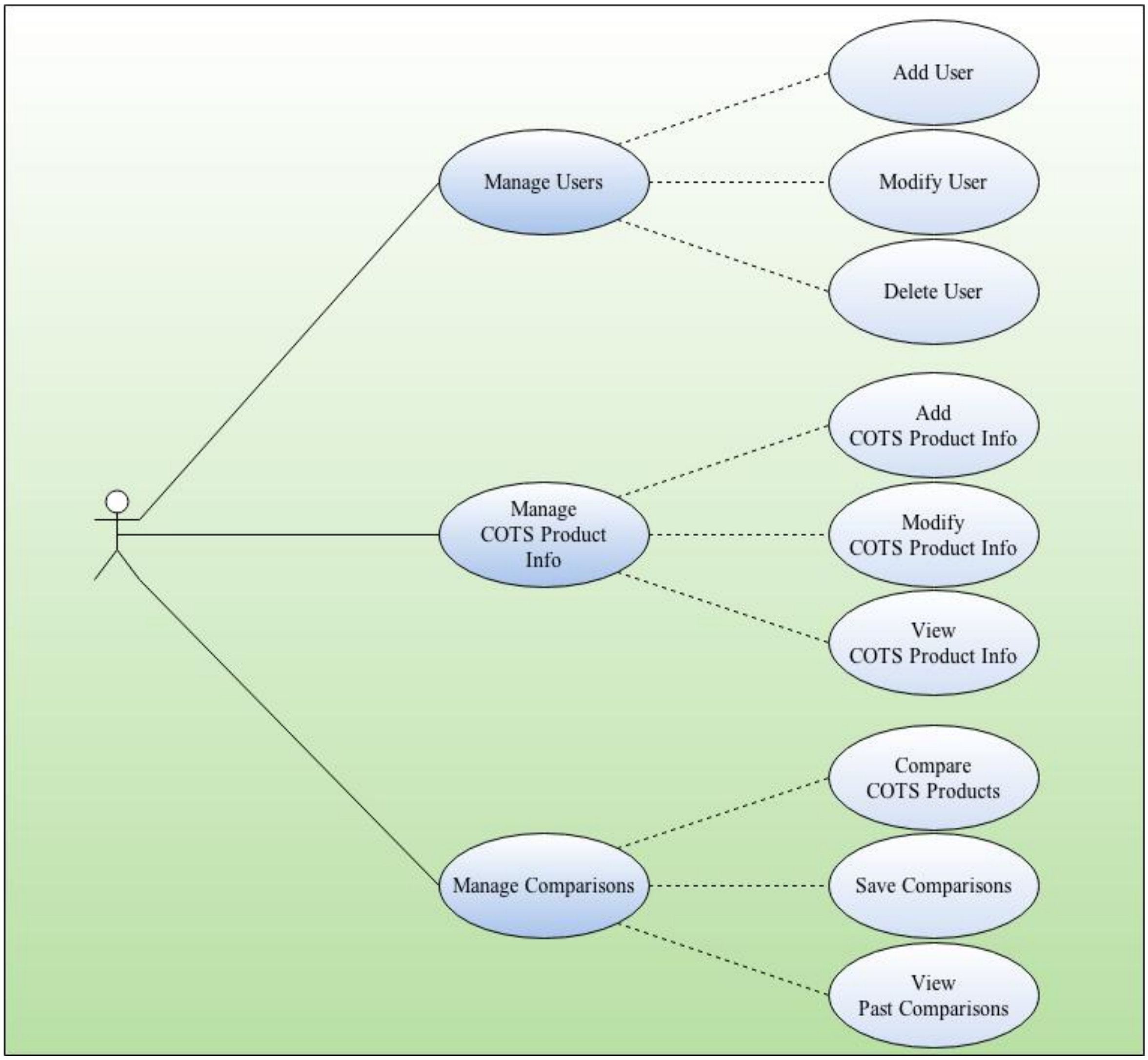

Figure 15: COTSEVAL Use-Case

4.1.2 COTSEVAL User Interface

Only the main functionalities along with the screenshots of main windows are shown in this section.

4.1.2.1 Main Menu

Once the user successfully logs into the system by providing the correct combination of a username and a password, the main menu is shown to the user. It has three buttons that redirect the user to the user management, COTS product or comparison area accordingly. The snapshot of the main window is shown in the figure 16 as under:

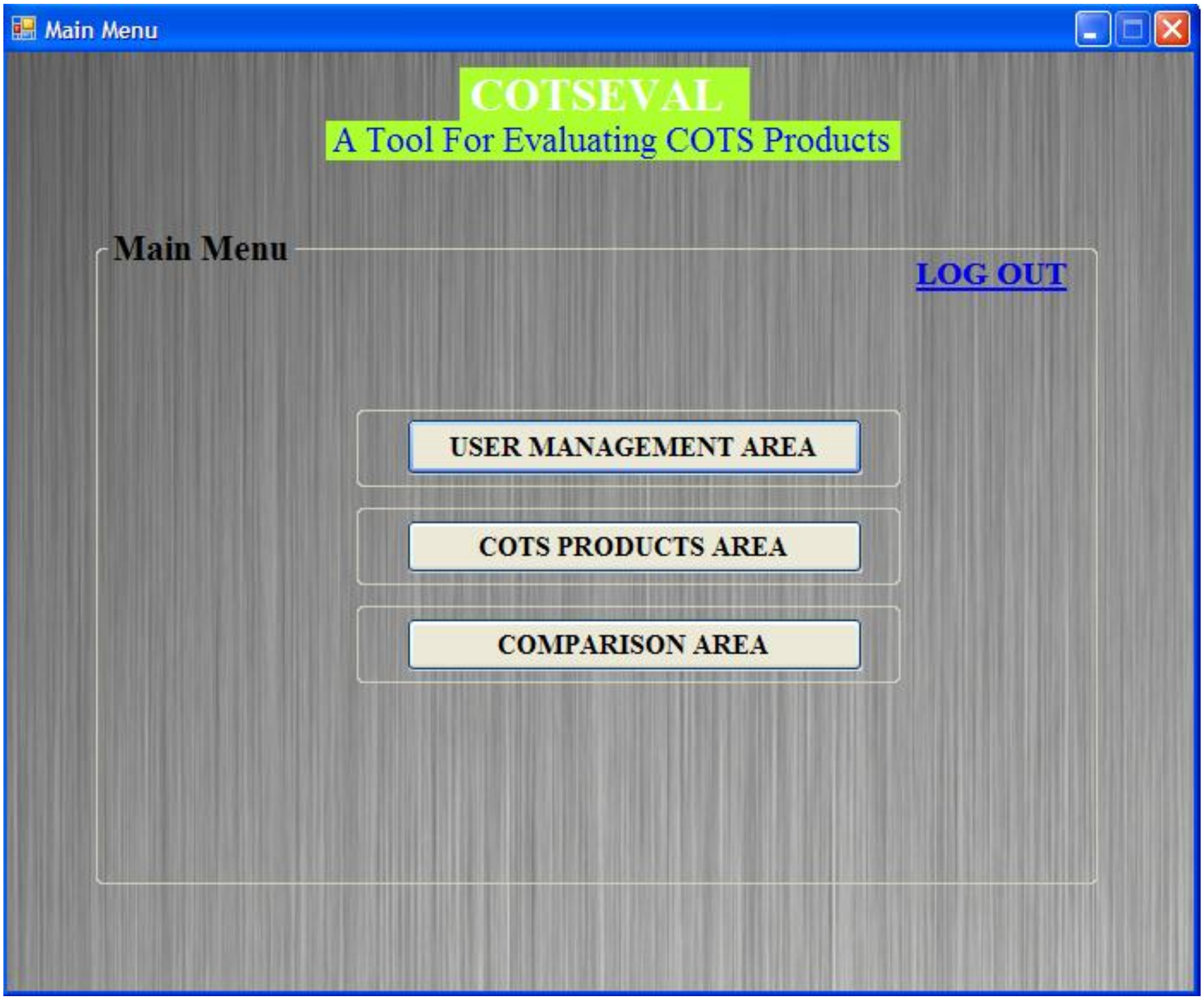

Figure 16: COTSEVAL Main Menu

4.1.2.2 User Management

Upon clicking on the user management area from the main menu, the user is shown a new window where user can select any of the available options, which are: to add a new user, modify user information and delete an existing user. The user-management area is shown in the Figure 17.

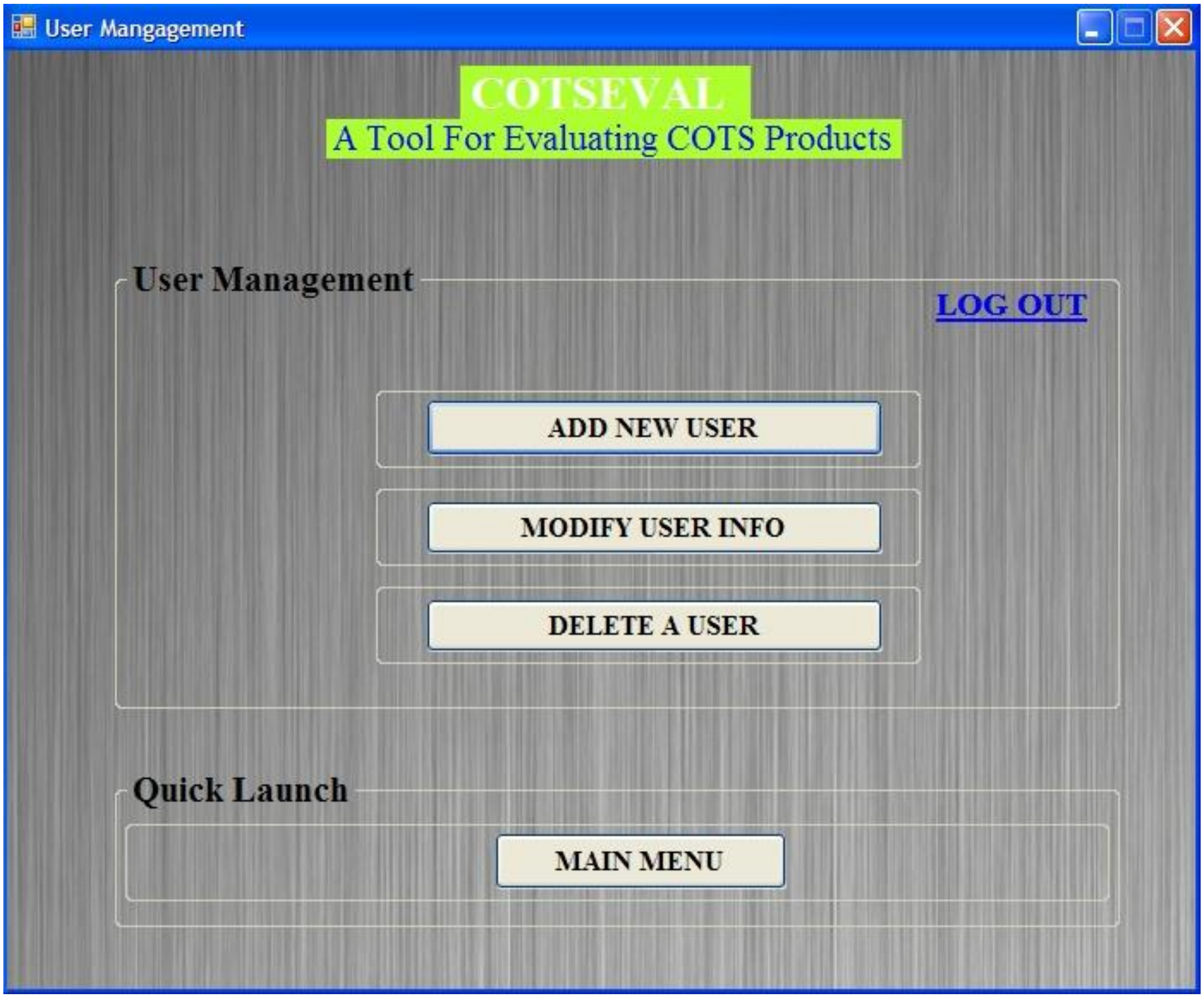

Figure 17: COTSEVAL User Management

4.1.2.3 COTS Management Area

In COTS management area a user can add information about a new COTS product which will be saved in the database, view information about the products saved in the past and modify the products information which will update the record in the database. The screen-capture of the COTS management window is shown in figure-18. To be able to perform a comparison, it is important to first have information about the products that will be considered for a comparison. This area helps in keeping up-to-date information about all the products. User-friendliness of this tool enables the user to do so with only few clicks. The buttons enable the user to navigate between windows very easily.

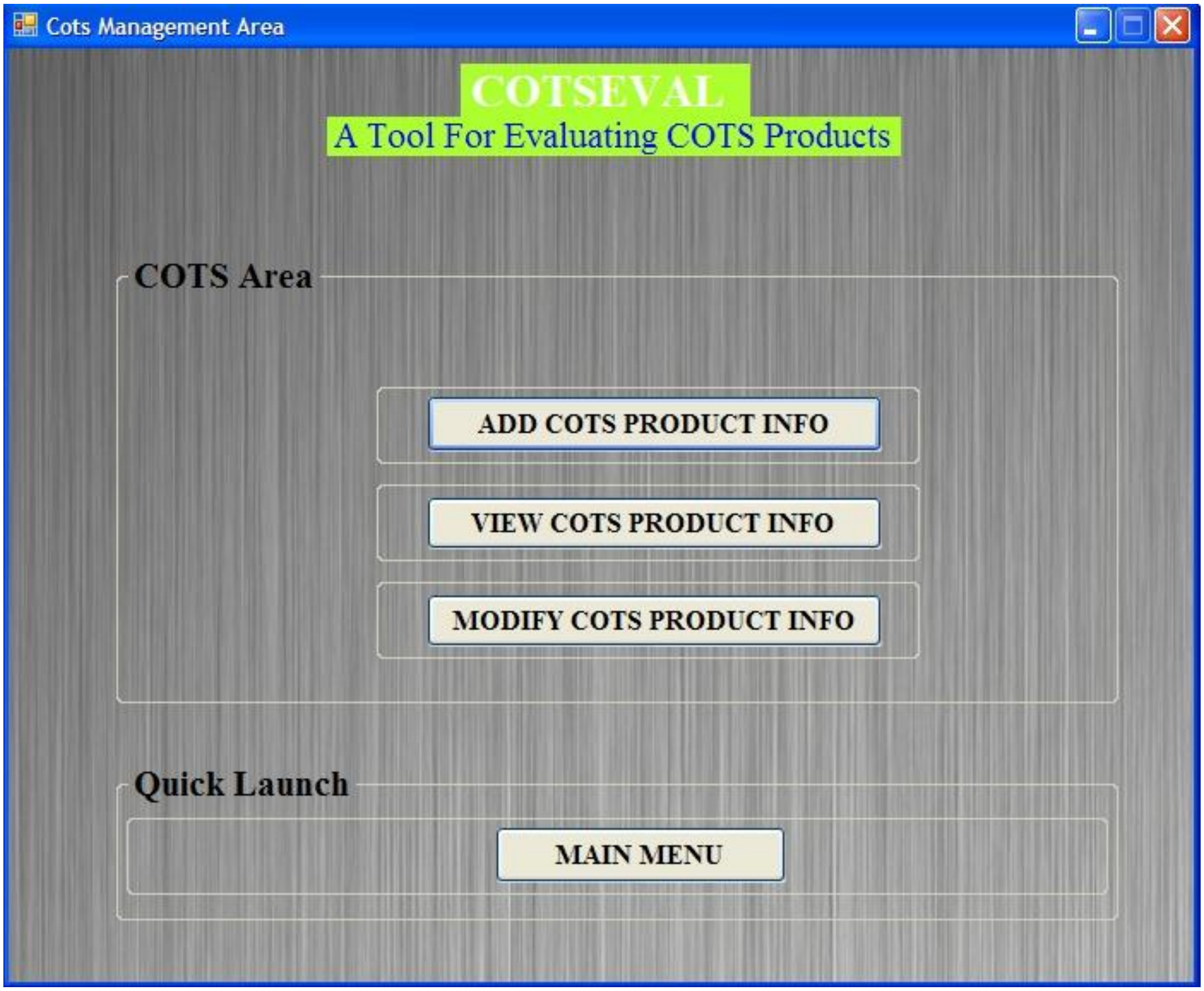

Figure 18: COTSEVAL COTS Area

4.1.2.4 Adding Product's Information

As shown in figure-19, a user has to add two kinds of information about each product i.e. General Information and Attributes Data. In general information the compulsory fields have an asterisk sign at the end of their name and rests are optional. In Attributes Data, user selects input values for all 18 attributes from combo-boxes that will be used in comparisons. The information about all these 18 attributes is mandatory. Once the information has been provided, the user has to click on "Save" button to add the entry in the database. The form is first validated and upon successful validation the information is entered as a new record. The "COTS ID" field uniquely identifies the products. In-case the user enters an ID that has been previously used, an appropriate error message is shown and user is asked to re-enter another ID. Besides that if a mandatory field's data is found to be missing, the user is shown an error message and asked to completely fill the form.

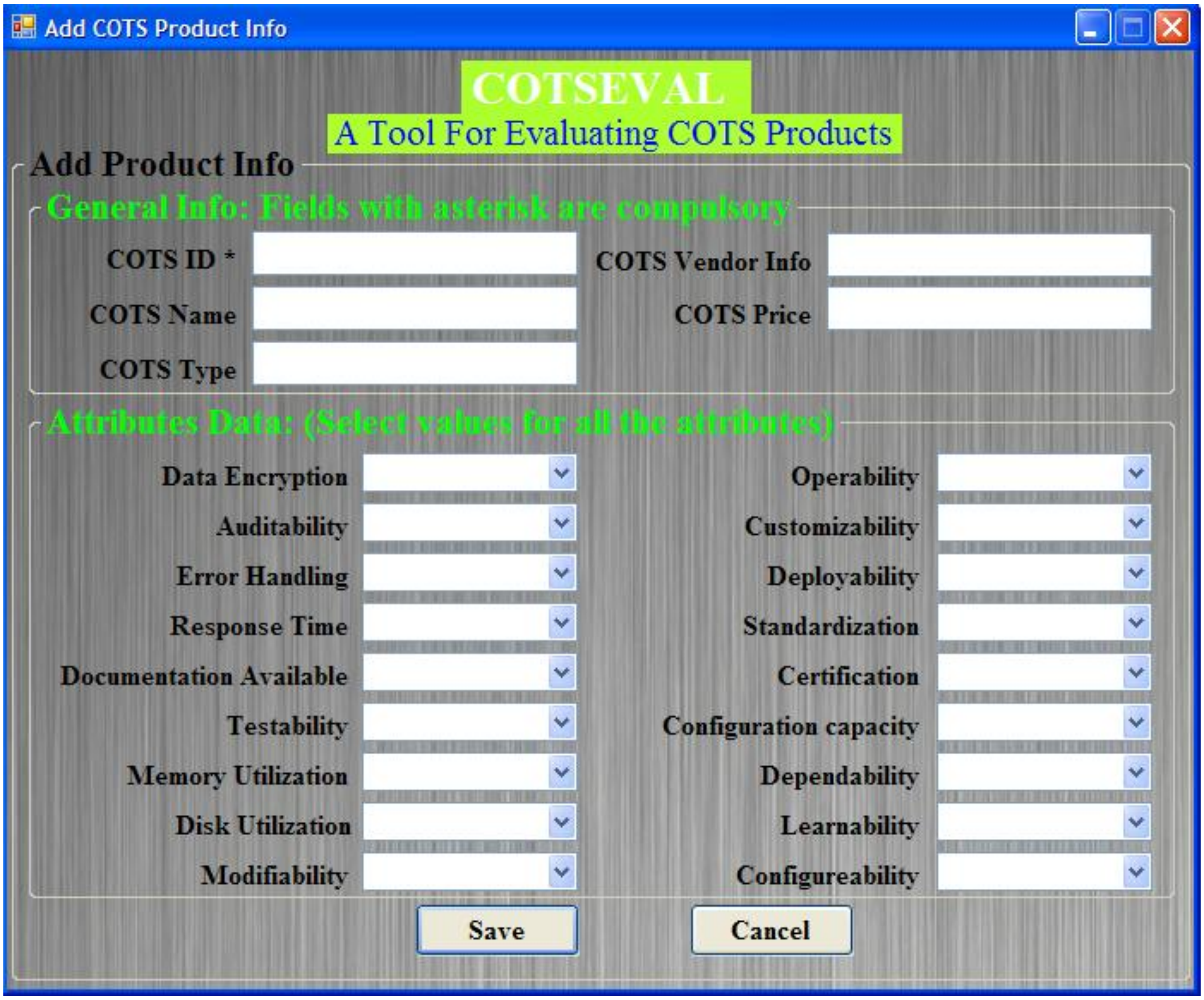

Figure 19: COTSEVAL Add Product's Info

4.1.2.5 Comparison Area

In comparison area, user has two options: make a comparison by selecting desired COTS products and view information about the comparison performed in the past by selecting the Comparison ID from the combo-box. Figure-20 shows the comparison area.

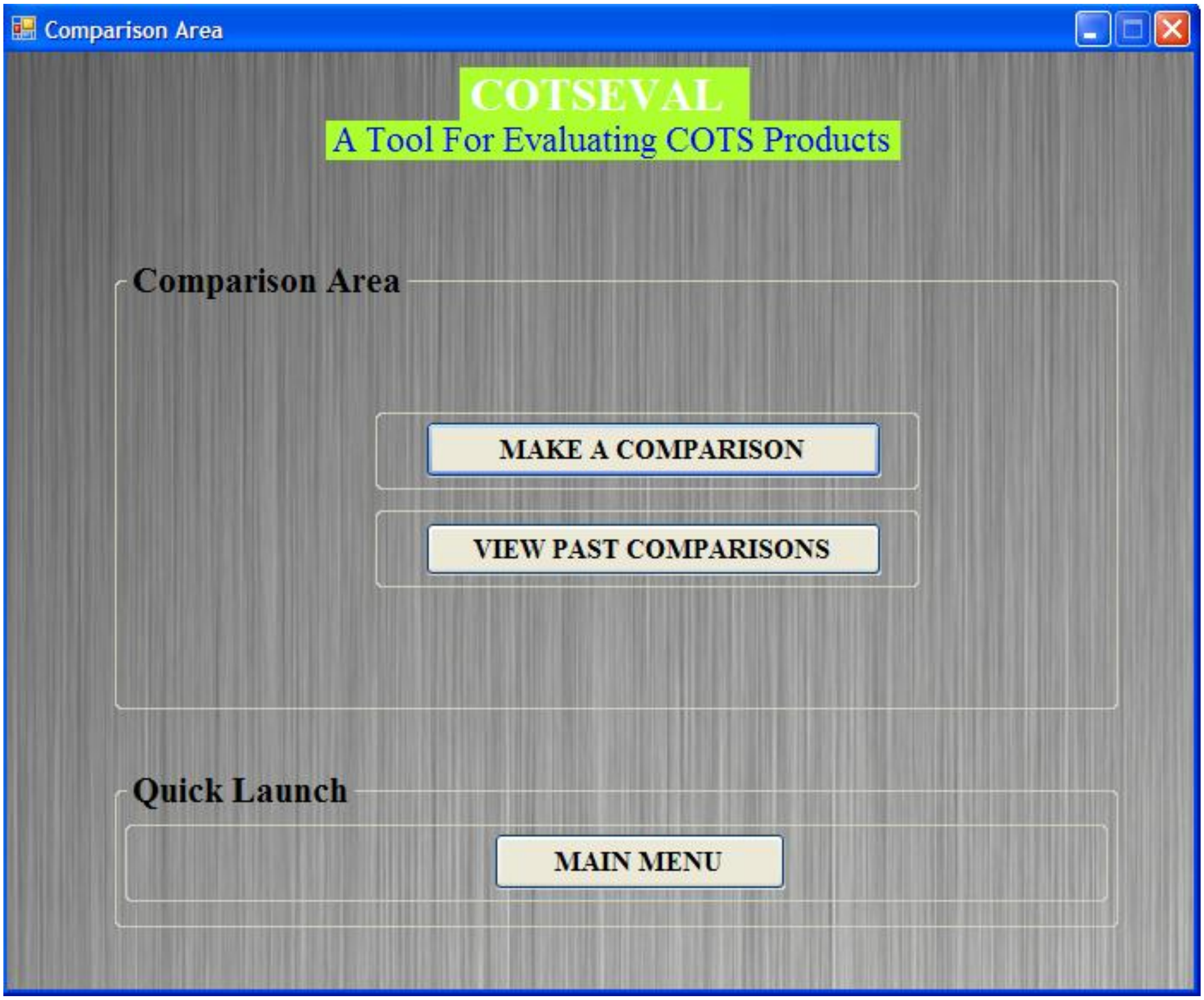

Figure 20: COTSEVAL Comparison Area

4.1.2.6 Perform a Comparison

To perform a comparison a user has to enter a comparison ID, select the checkboxes for the products he/she desires to compare and then click on "Done" button, figure 21-A. Upon clicking the button, the information about the comparison is saved in the database and same information and results are shown in another DataGridView. The results contain information about each product's class scores, class values and overall score and suitability of the product as shown in figure 21-B. This comparison and its result can be viewed in future by selecting the comparison ID from the Combo-Box in "View Past Comparison" window.

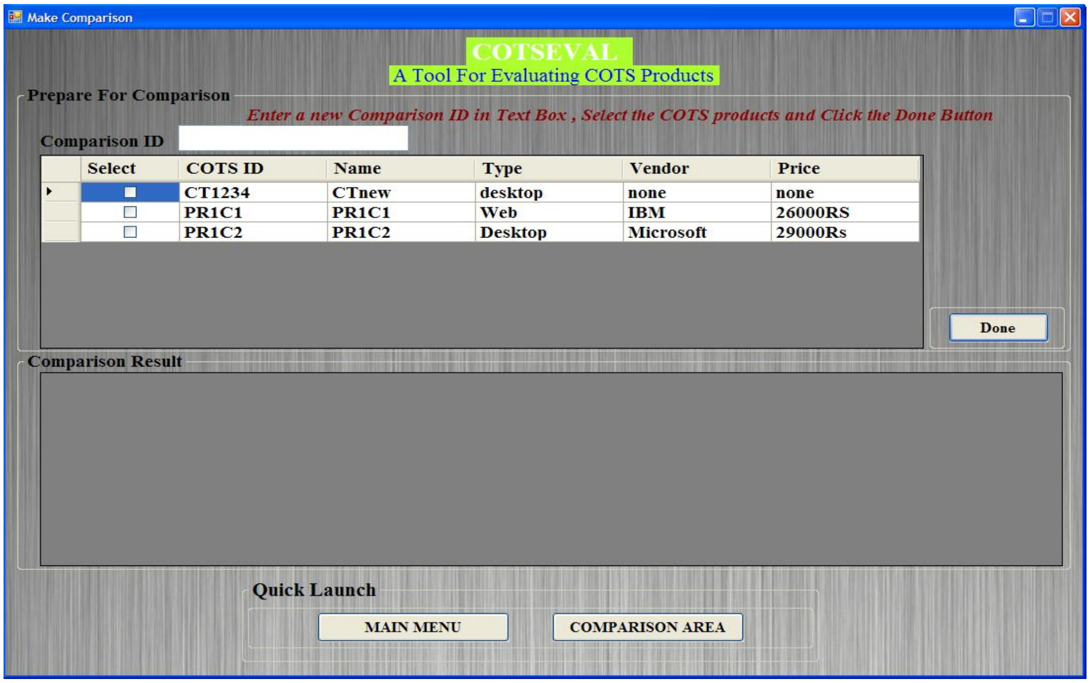

Figure 21-A: COTSEVAL Perform a Comparison

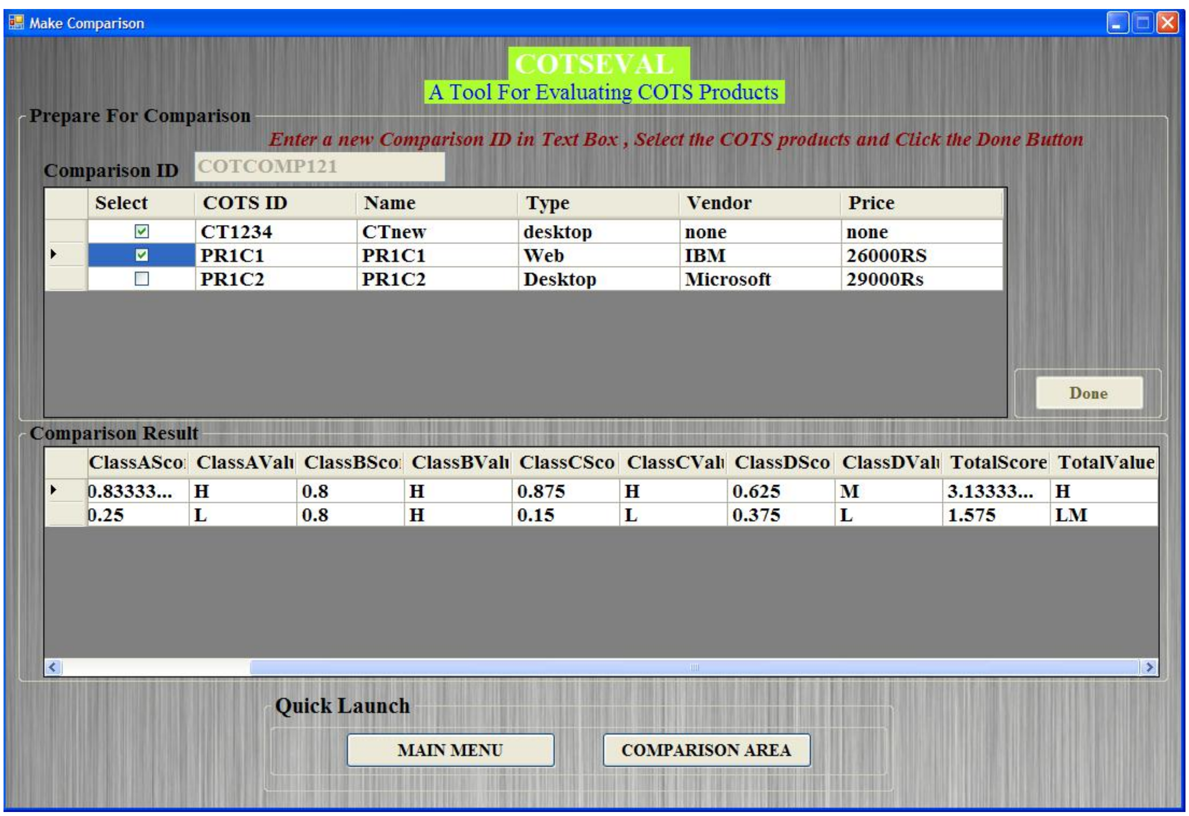

Figure 21-B: COTSEVAL Perform a Comparison

4.2 Comparison and Results

The proposed approach has been validated in three ways:

- **Qualitative Comparison**
- **Validation Part-1 (Data-Set)**
- **Validation Part -2 (Survey)**

These are discussed in detail in the following sub-sections.

4.2.1 Qualitative Comparison

The following factors are considered in qualitative comparison:

- **Stages:** Number of stages in the approach.
- **Evaluation:**
    - **ET:** Evaluation technique used in the approach.
    - **ES:** Evaluation strategy.
- **MC:** Is multiple criteria used for evaluation i.e. more than one type of criteria.
- **RD:** Are user requirements discussed and mapped to evaluation technique.
- **NFRs:** Were non-functional requirements considered in evaluation. Four types of NFRs:
    - **QA:** Quality attributes
    - **AA:** Architecture attributes
    - **DA:** Domain attributes
    - **OA:** Organizational attributes
- **SDD:** Were stages described in detail and guidelines were provided to perform each task.
- **ST:** Is there any software tool accompanying the approach to assist in evaluation.

The comparison is shown in the table 6 as under:

Table 6: Qualitative Comparison

<table>
<tr><th rowspan="2">Approaches</th><th rowspan="2">Stages/ Steps</th><th colspan="2">Evaluation</th><th rowspan="2">MC</th><th rowspan="2">RD</th><th colspan="4">NFRs</th><th rowspan="2">SDD</th><th rowspan="2">ST</th></tr>
<tr><th>ET</th><th>ES</th><th>QA</th><th>AA</th><th>DA</th><th>OA</th></tr>
<tr><td rowspan="2">IROTS [14]</td><td rowspan="2">09</td><td rowspan="2">DNP</td><td rowspan="2">DNP</td><td rowspan="2">DNP</td><td rowspan="2">N</td><td colspan="4">Y</td><td rowspan="2">N</td><td rowspan="2">N</td></tr>
<tr><td>Y</td><td>Y</td><td>Y</td><td>Y</td></tr>
<tr><td rowspan="2">W-Shaped [15]</td><td rowspan="2">05</td><td rowspan="2">ANP</td><td rowspan="2">***</td><td rowspan="2">Y</td><td rowspan="2">DNP</td><td colspan="4">Y</td><td rowspan="2">PNC</td><td rowspan="2">Y</td></tr>
<tr><td>DNP</td><td>DNP</td><td>DNP</td><td>DNP</td></tr>
<tr><td rowspan="2">CARE [22]</td><td rowspan="2">05</td><td rowspan="2">Ranking</td><td rowspan="2">PF</td><td rowspan="2">DNP</td><td rowspan="2">Y</td><td colspan="4">Y</td><td rowspan="2">N</td><td rowspan="2">N</td></tr>
<tr><td>DNP</td><td>DNP</td><td>DNP</td><td>DNP</td></tr>
<tr><td rowspan="2">CEP [19]</td><td rowspan="2">05</td><td rowspan="2">Weighted Averages</td><td rowspan="2">PF</td><td rowspan="2">DNP</td><td rowspan="2">N</td><td colspan="4">Y</td><td rowspan="2">PNC</td><td rowspan="2">N</td></tr>
<tr><td>DNP</td><td>DNP</td><td>DNP</td><td>DNP</td></tr>
<tr><td rowspan="2">TAES-COTS</td><td rowspan="2">05</td><td rowspan="2">AHP + RBR+ Priorities</td><td rowspan="2">PF</td><td rowspan="2">Y</td><td rowspan="2">Y</td><td colspan="4">Y</td><td rowspan="2">Y</td><td rowspan="2">Y</td></tr>
<tr><td>Y</td><td>Y</td><td>Y</td><td>Y</td></tr>
<tr><td colspan="4">PF - Progressive filtration
RBR - Rule-based Reasoning
ANP - Analytical Network Process
*** - Uncertain Strategy</td><td colspan="8">PNC - Partially and Not Completely
DNP - Detail Not Provided
N - No
Y - Yes</td></tr>
</table>

4.2.2 Validation Part-1 (Data Set)

To determine the accuracy of the detailed evaluation technique that is conducted in step-4 of the proposed approach (described in section 3.2.4), the COTSEVAL was populated with hypothetical data of 32 products given in Table-B1, Appendix-B. The same data set was shared with 15 professionals working in various private and public sector organizations. Five evaluation scenarios were considered, each considering sub-sets with randomly selected products from the main data set.

Few conditions were established prior to the comparison:

- All the participants and COTSEVAL will have same product's information i.e. 18 attributes data, in any given scenario.
- All the participants shall consider the same level of priority and distribution of classes for a fair comparison.

- The participants may use any technique to evaluate products provided that condition number two stands true.
- The final product selected by majority of participants will be considered the optimal product.
- Participants can choose more than one product.

Details of the comparison are shown in table 7.

Table 7: Comparison Information

Scenarios	No of Products	Products Involved
SC-1*	05	CT49600 CT42200 CT19400 CT78990 CT67760
SC-2	08	CT85634 CT42200 CT85600 CT57400 CT90092 CT46940 CT65438 CT90800
SC-3	11	CT84000 CT57400 CT73833 CT85634 CT85670 CT78990 CT89300 CT25630 CT35090 CT94500 CT98600
SC-4	16	CT25630 CT46940 CT57400 CT77820 CT98600 CT84000

		CT84676 CT85634 CT90092 CT90800 CT94500 CT28453 CT94679 CT35090 CT38500 CT42200
SC-5	20	CT84000 CT84676 CT85634 CT85670 CT38500 CT42200 CT46940 CT67760 CT95700 CT77820 CT78990 CT49600 CT57400 CT89300 CT90092 CT90800 CT25630 CT35090 CT94500 CT94679
**SC-1 to SC-5 represents the five scenarios*		

The results of the comparisons are given in table 8.

Table 8: Comparison Results

Scenarios	Participant's Selection		COTSEVAL's Selection
	Product Selected	No of Participants	
SC-1	CT19400	15	CT19400

SC-2	CT65438	12	CT65438 & CT85600
	CT65438 & CT85600	3	
SC-3	CT73833	13	CT73833
	CT98600 & CT78990	01	
	CT78990	01	
SC-4	CT28453	13	CT28453
	CT28453 & CT98600	01	
	CT28453 & CT94500	01	
SC-5	CT95700	15	CT95700

In each scenario random products were selected from the data-set and in each case our approach showed 100 percent accuracy and selected the optimal candidates. Figure 22 shows the percentage of exact and majority match for each scenario.

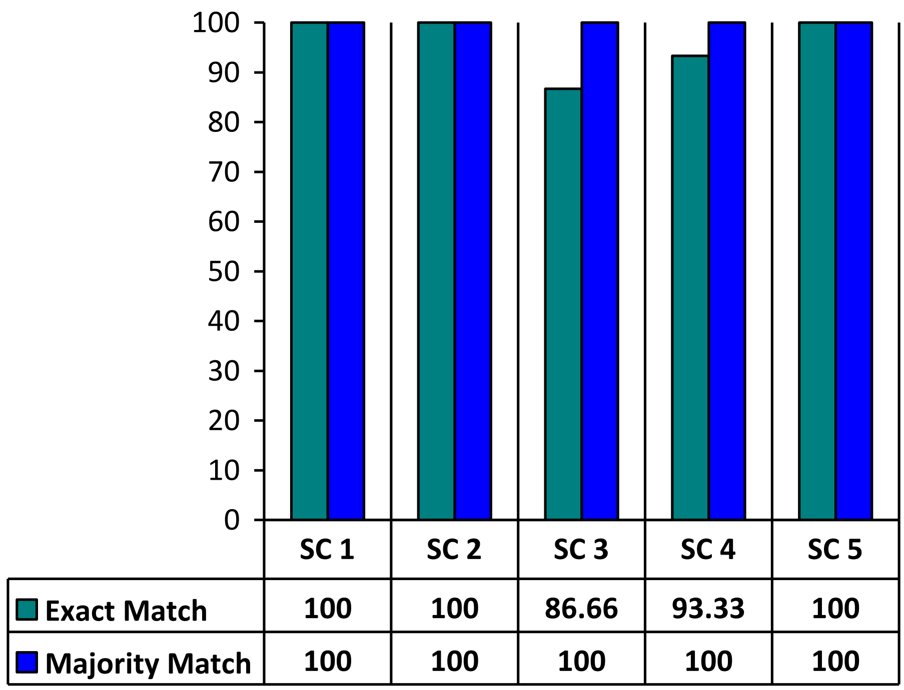

Figure 22: Detailed Evaluation Technique's Accuracy

The advantage of this validation is that not only the whole evaluation technique was validated but also the 81rules were proved to be accurate. The time and effort required to complete the evaluation were other factor that were witnessed to be very less using COTSEVAL as compared to the manual evaluation carried out by the participants.

4.2.3 Validation Part-2 (Survey)

The whole approach and detailed evaluation technique were also validated by a survey. A questionnaire was prepared and participants of the survey were asked to answer 21 questions. The survey form can be seen in appendix-B. The form has three sections: General Software evaluation & selection related questions, TAES-COTS approach related questions and TAES-COTS's detailed evaluation related questions. The questions asked in the questionnaire helped in measuring various factors. The factors and the questions whose inputs were used to measure them are shown in table-9 below:

Table 9: Factors Measured in Survey

Whole TAES-COTS Approach		
S-No	**Factors**	**Questions #**
1.	Thoroughness	06 & 07
2.	Effort	08, 12 & 14
3.	Simplicity	09 & 15
4.	Accuracy	11 & 13
5.	Practicality	10
TAES-COTS Detailed Evaluation		
S-No	**Factors**	**Question #**
1.	Completeness	20
2.	Effort	16
3.	Time Consumption	18
4.	Simplicity	19
5.	Accuracy	17
6.	Practicality	21

20 professionals from various private and public sector organizations participated in the survey. The results of the survey are shown in table 10:

Table 10: Survey Results

<table>
<tr><th>Question #</th><th colspan="4">Answers</th></tr>
<tr><td colspan="5">General Selection Related Questions</td></tr>
<tr><td rowspan="2">1.</td><td colspan="2">Yes</td><td colspan="2">No</td></tr>
<tr><td colspan="2">00</td><td colspan="2">20</td></tr>
<tr><td>2.</td><td colspan="4">• Cost
• User Ratings
• Functionalities</td></tr>
<tr><td>3.</td><td colspan="4">Evaluation & Requirement gathering</td></tr>
<tr><td rowspan="2">4.</td><td>Yes</td><td>No</td><td colspan="2" rowspan="2">Yes Reason:
Better results
Other Reason:
Depends on the cost of acquisition</td></tr>
<tr><td>17</td><td>00</td></tr>
<tr><td rowspan="2">5.</td><td colspan="2">Yes</td><td colspan="2">No</td></tr>
<tr><td colspan="2">20</td><td colspan="2">00</td></tr>
<tr><td colspan="5">TAES-COTS Related Questions</td></tr>
<tr><td rowspan="2">6.</td><td colspan="2">Yes</td><td colspan="2">No</td></tr>
<tr><td colspan="2">20</td><td colspan="2">00</td></tr>
<tr><td rowspan="2">7.</td><td colspan="2">Yes</td><td colspan="2">No</td></tr>
<tr><td colspan="2">20</td><td colspan="2">00</td></tr>
<tr><td rowspan="2">8.</td><td colspan="2">Low</td><td>Medium</td><td>High</td></tr>
<tr><td colspan="2">15</td><td>05</td><td>00</td></tr>
<tr><td rowspan="2">9.</td><td>Yes</td><td>No</td><td colspan="2" rowspan="2">Comment:
01 - Gathering data for attributes
No - Initial screening
01 – Requirement gathering</td></tr>
<tr><td>17</td><td>01</td></tr>
</table>

10.	**Yes**	**No**	**Other:**
	16	01	**01** - Also makes it lengthy **01** – 70% **01** – 80%
11.	**Yes**		**No**
	20		00
12.	**Yes**	**No**	**Other:**
	17	00	**03** - 80 to 90%
13.	**Yes**	**No**	**Other**
	17	00	**02** – 90 % **01** – 80%
14.	**Yes**	**No**	**No Reason:**
	18	01	It will be time consuming **Other:** 01 - Maybe 80%
15.	**Yes**	**No**	**Yes Reasons:**
	18	00	• It will be efficient • It is easy to implement • It is thorough **Other**: **02** - May be 80%
TAES-COTS's Detailed Evaluation Related Questions			
16.	**Yes**	**No**	**Other**
	16	00	**02** - 80% **02 -** 90%
17.	**Yes**		**No**
	20		00

18.	**Low**		**Medium**	**High**
	17		03	00
19.	**Yes**		**No**	
	20		00	
20.	**Yes**		**No**	
	20		00	
21.	**Yes**	**No**	**Other**	
	17	01	**02** – 70%	

Based on the answers given by the participants, the factors discussed in table 9 were measured and the results are shown in figures 23 and 24. Figure 23 shows the results for the detailed evaluation technique and figure 24 shows results for the whole TAES-COTS approach.

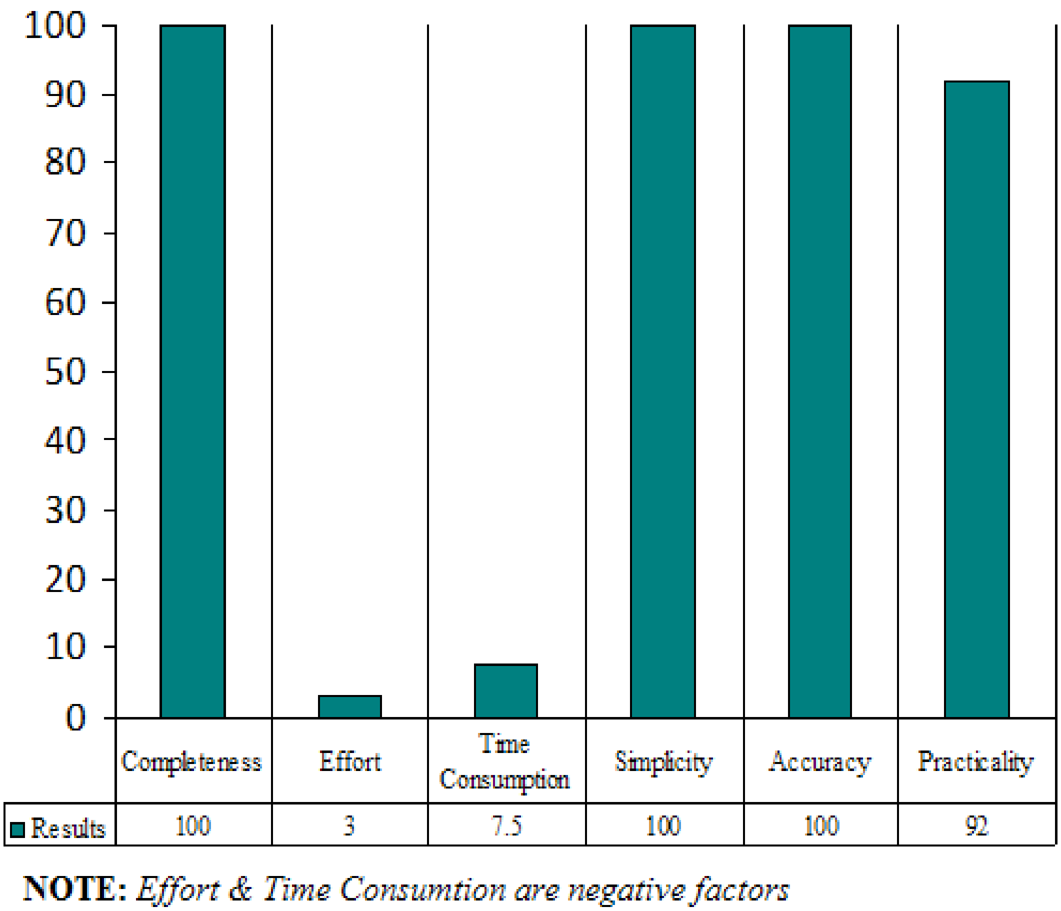

Figure 23: Survey Results - Detailed Evaluation Technique

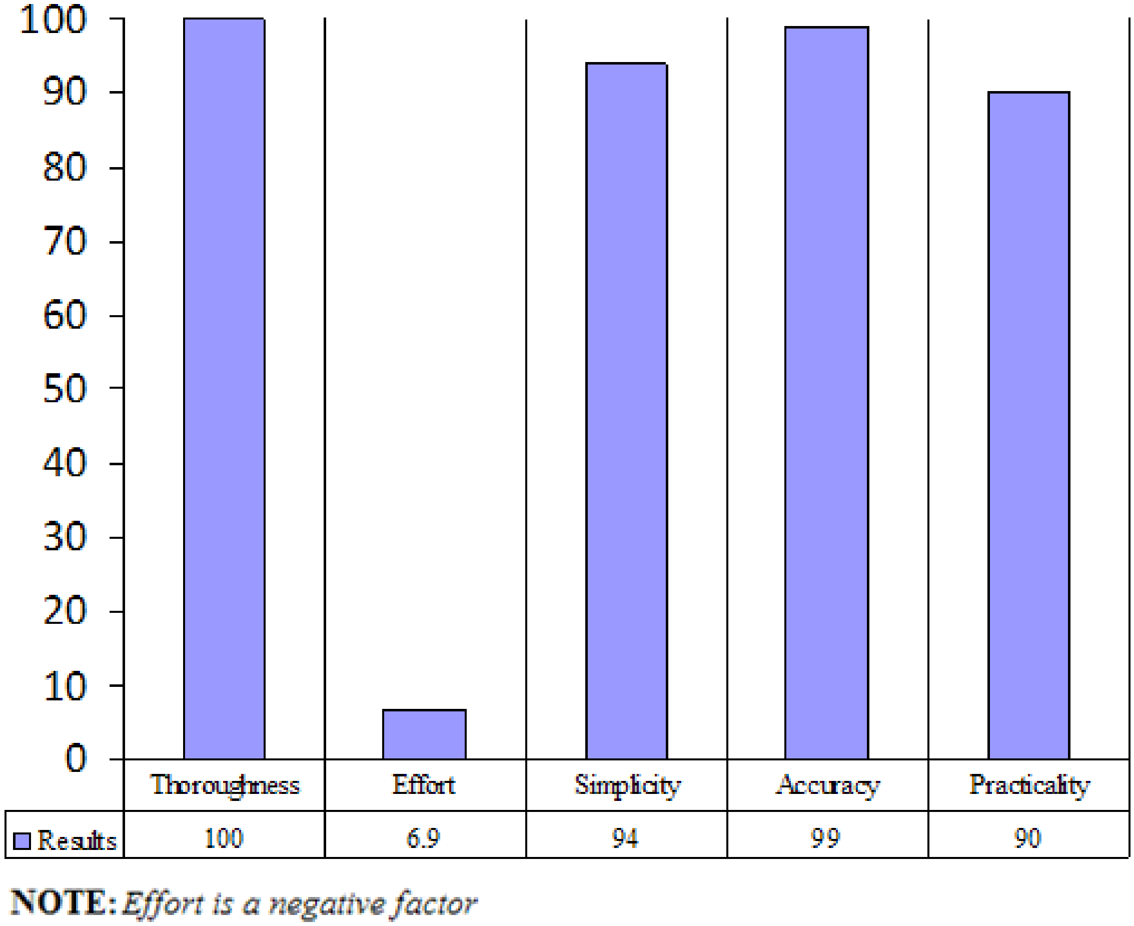

Figure 24: Survey Results – TAES-COTS Approach

Summary

COTSEVAL is the software tool that assists the users in carrying out the detailed evaluation discussed in previous chapter. This chapter discusses the COTSEVAL's user interface. Besides that it also discusses the validation of the proposed approach.

Chapter 5 Conclusions & Future Work

This chapter concludes this research work and discusses the probable research areas that can be considered in future. Section 6.1 contains the summary of the research and section 6.2 comprises the discussion about the future work.

5.1 Conclusion

With an increase in the use of packaged software and availability of multiple products to serve one purpose, it has become necessary to be able to choose the right option. The main motive behind usage of such ready-made software is to save time and development cost. If chosen correctly, the product will rightly serve its purpose. To choose the correct product, one must evaluate them thoroughly. There are a number of evaluation approaches present in literature. Not many of them are used practically, reasons varying from their complexity to their not being detailed enough.

TAES-COTS is an approach that tries to overcome the limitations found in other approaches with the objective of being thorough and at the same time simple enough to be practically implementable. In its five steps, the evaluators are properly guided to perform specific tasks and also instructed as how to perform them. Using this approach each product will be rigorously evaluated and only the most suitable candidate will make it till the end. To save the evaluators some time, this approach is accompanied by a prototype tool "COTSEVAL" that helps the examiners conduct the NFRs based evaluation. Evaluation starts with a number of products whose evaluation would otherwise be confusing and impossible, but following this approach will lead to selection of a single well-suited product that can be trusted to properly perform its functionalities and maintain the level of performance expected from it.

5.2 Future Work

There is a lot that can be done to better the evaluation of software products. Although a number of selection and evaluation approaches have being proposed in the last 2 decades but this area of research is still developing with not having a single formal approach that completely address all issues that arise in selection process.

In future we intend to address the issue of evaluation of multiple products simultaneously which fit together and work in combination to achieve desired functionalities. This multi-product selection approach will be based on puzzle assembly strategy that will be applicable in situations where a single product is not sufficient enough to meet the requirements and should be used together with other product or products.

References

[1]. Maurizio Morisio, Nancy Sunderhaft. Commercial-Off-The-Shelf (COTS): A Survey, A DACS State-of-the-Art Report. Prepared for: Air Force Research Laboratory – Information Directorate (AFRL/IFED), December 2000

[2]. E. Kesseler, Assessing COTS Software in a Certifiable Safety-Critical Domain. Information Systems Journal (18) 2008

[3]. M. Morisio, C.B. Seaman, V.R. Basili, A.T. Parra, S.E. Kraft, S.E. Condon. COTS-Based Software Development: Processes and Open Issues. Journal of Systems and Software, Volume 61 Issue 3, April 2002

[4]. Abdallah Mohamed, Guenther Ruhe, Armin Eberlein. COTS Selection: Past, Present, and Future. Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'07), 2007

[5]. Feras Tarawneh, Fauziah Baharom, Jamaiah Hj. Yahaya, and Faudziah Ahmad. Evaluation and Selection COTS Software Process: The State of the Art. The Society of Digital Information and Wireless Communications (SDIWC): International Journal of New Computer Architectures and their Applications (IJNCAA), 2011.

[6]. Andrew Stellman, Jennifer Greene. Applied Software Project Management, O'Reilly Media, ISBN 978-0-596-00948-9, 2005

[7]. Ljerka Beus-Dukic, Non-Functional Requirements for COTS Software Components, Proceedings of ICSE workshop on COTS Software, 2000.

[8]. Marek J. Druzdzel and Roger R. Flynn. Decision Support Systems. In Encyclopedia of Library and Information Science, Third Edition, Marcia J. Bates and Mary Niles Maack (eds.), Taylor & Francis, Inc., New York, 16 February 2010.

[9]. Daniel J. Power, Decision Support Systems: Concepts and Resources for Managers ISBN: 156720497X, 2002

[10]. Thomas L. Saaty, Kirti Peniwati. Group Decision Making: Drawing out and Reconciling Differences. Pittsburgh, Pennsylvania: RWS Publications. ISBN: 978-1-888603-08-8, 2008.

[11]. Thomas L. Saaty. Decision Making for Leaders: The Analytic Hierarchy Process for Decisions in a Complex World. Pittsburgh, Pennsylvania: RWS Publications. ISBN 0-9620317-8-X, 2008

[12]. Jocelyn Ireson-Paine. What is a rule-based system? Available at: http://www.jpaine.org/students/lectures/lect3/node5.html, last accessed on May 07th, 2014

[13]. P. C. Jha, Vikram Bali, Sonam Narula, Mala Kalra. Optimal Component Selection Based on Cohesion and Coupling for Component-Based Software System. Advances in Intelligent Systems and Computing Volume 236, 2014

[14]. M. Shakeel Faridi & Zahid Javed, M. Haris Abid, Mudassar Ahmed, Dr. Md Asri Bin Ngadi. IROTS: A Proposed COTS Evaluation & Selection Methodology for Component Based Software Engineering in Under-Development Countries. 2nd International Conference on Advances in Computer Science and Engineering, 2013

[15]. Vinay, Manoj Kumar and Prashant Johri, W-Shaped Framework for Component Selection and Product Development Process, World Applied Sciences Journal 31 (4), 2014

[16]. Thomas Neubauer, Christian Stummer, Interactive Decision Support for Multiobjective COTS Selection. Proceedings of the 40th Hawaii International Conference on System Sciences, 2007

[17]. J. Kontio, OTSO: A Systematic Process for Reusable Software Component Selection, CSTR-3478, 1995.

[18]. M. Morisio and A. Tsoukias, "IusWare: a methodology for the evaluation and selection of software products," IEE Software Engineering vol. 144, 1997.

[19]. Barbara Cavanaugh Phillips and Susan M. Polen Add Decision Analysis to Your COTS Selection Process The Journal of Defense Software Engineering, Software Technology Support Center Crosstalk, 2002.

[20]. Alves Carina and Jaelson Castro. CRE: A systematic method for COTS components selection. XV Brazilian Symposium on Software Engineering (SBES), 2001

[21]. Michael Ochs, Dietmar Pfahl, Gunther Chrobok-Diening, Beate Nothhelfer-Kolb, A COTS acquisition process: Definition and application experience ISERN Report, 2000.

[22]. Chung Lawrence, Kendra Cooper, Sam Courtney COTS-Aware requirements engineering: The CARE process. Proceedings of the 2nd International Workshop on Requirements Engineering for COTS Components (RECOTS), 2004

[23]. Douglas Kunda and Laurence Brooks, Applying social-technical approach for COTS selection, Proceedings of the 4th UKAIS Conference, 1999

[24]. Santiago Comella-Dorda, John Dean, Grace Lewis, Edwin Morris, Patricia Oberndorf, Erin Harper, A process for COTS software product evaluation, 2002

[25]. Abdallah Mohamed, Guenther Ruhe, Armin Eberlein Mohamed, Decision support for handling mismatches between COTS products and system requirements, Commercial-off-the-Shelf (COTS)-Based Software Systems, ICCBSS'07, Sixth International IEEE Conference, 2007

[26]. Sallie Gregor, Joseph Hutson, and Colleen Oresky, Storyboard process to assist in requirements verification and adaptation to capabilities inherent in COTS, COTS-Based Software Systems, Springer Berlin Heidelberg, 2002

[27]. Hamdy Ibrahim, Abdel-Halim H. Elamy, Behrouz H. Far, Armin Eberlein, UnHOS: A Method for Uncertainty Handling in Commercial Off-The-Shelf (COTS) Selection, International Journal of Energy, Information and Communications Vol. 2, Issue 3, 2011

[28]. Patricia K. Lawlis, Kathryn E. Mark, Deborah A. Thomas and Terry Courtheyn, A formal process for evaluating COTS software products, 2001

[29]. Carina Alves and Anthony Finkelstein, Investigating conflicts in COTS decision-making. International Journal of Software Engineering and Knowledge Engineering, Volume13, 2003

[30]. TomWanyama and Behrouz Homayoun Far, Towards providing decision support for COTS selection, Electrical and Computer Engineering, Canadian Conference on IEEE, 2005.

[31]. Rikard Land, Laurens Blankers, Michel Chaudron and Ivica Crnkovic, COTS selection best practices in literature and in industry, High Confidence Software Reuse in Large Systems, Springer Berlin Heidelberg, 2008

[32]. Tom Wanyama and Behrouz H. Far, An Empirical Study To Compare Three Methods For Selecting Cots Software Components, International Journal of Computing and ICT Research, Volume 2, 2008

[33]. Zahid Javed, Ahsan Raza Sattar & Muhammad Shakeel Faridi, Unsolved Tricky Issues on COTS Selection and Evaluation, Global Journal of Computer Science and Technology, Volume 12 Issue 10, 2012

[34]. M. Morisio, C.B. Seaman, A. T. Parra, V.R. Basili, S.E. Kraft, S.E. Condon. Investigating and improving a COTS-based software development, Proceedings of the 22nd international conference on Software engineering ACM, 2000

[35]. Systems Engineering Process Office, Commercial-Off-the-Shelf (COTS) Evaluation, Selection, and Qualification Process, PR-SPE-05 V1.0, Code 212, Space and Naval Warfare Systems Center, 2002

[36]. Jyrki Kontio. A case study in applying a systematic method for COTS selection, Proceedings of the 18th International Conference on Software Engineering, 1996

[37]. Alexandre Alvaro, Eduardo Santana de Almeida, Silvio Romero de Lemos Meira. Quality attributes for a component quality model." 10th WCOP/19th ECCOP, 2005

[38]. Alexandre Alvaro, Eduardo Santana de Almeida, Silvio Lemos Meira. A software component quality model: A preliminary evaluation. Proceedings of the 32nd EUROMICRO Conference on Software Engineering and Advanced Applications (EUROMICRO-SEAA'06), 2006

Appendix-A

Step-3 Evaluation Template

Table A-1: Functional Requirements Based Evaluation

S-No	Functional Requirements (Mandatory)	C1*	C2	C3	C4	C5	C6	C7	. . .	Cn
1		Yes**	Yes	Yes	Yes	No	Yes	No	. . .	…
2		Yes	Yes	Yes	No	No	Yes	No	. . .	…
3		Yes	Yes	Yes	No	Yes	Yes	No	. . .	…
4		Yes	Yes	Yes	Yes	No	Yes	No	. . .	…
	. . .	. . .	. . .						. . .	. . .
	Functional Requirements (Optional)									
1		Yes	Yes	Yes	Yes	Yes	Yes	No	. . .	...
2		Yes	Yes	No	Yes	Yes	Yes	Yes	. . .	…
3		No	No	No	No	Yes	Yes	No	. . .	…
	. . .	. . .	. . .						. . .	. . .
	Budget Constraint (Price Within the limit?)	Yes	Yes	Yes	Yes	No	Yes	No		
Final Verdict		S***	S	S	R	R	S	R	. . .	…

****   C1, C2, …, Cn represent COTS products***
*****   Yes/No represent the presence or absence of a requirement***
****** S/R represent the status i.e. selected or rejected***

Step-5 Evaluation Template

Table A-2: Vendor Information Based Evaluation

S-No	Criteria	C1	C1 Score	C2	C2 Score	C3	C3 Score
1	**Time in market**	2year	0	10years	1	5years	0
2	**Number of Products**	15	1	10	0	5	0
3	**Number of successful products**	12	1	5	0	3	0
4	**Word on street**	Positive	1	Negative	0	Positive	1
5	**Organization size**	Large	1	Medium	0	Small	0
6	**Support Available**	Yes	1	Yes	1	Yes	1
7	**Past Experience**	none	0	Yes(-ve)	0	Yes(+ve)	1
	. . .	. . .		. . .			
	Total Score						
	Final Rank	1st - Selected		3rd		2nd	

Appendix-B

Data Set

Table B-1: Validation Data Set

Product ID	ATTRIBUTE VALUES (18 Attributes)																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
CT19400	0.6	1.0	1.0	1.0	0.9	1.0	0.7	0.9	1.0	0.6	1.0	1.0	0.8	0.8	0.9	1.0	1.0	0.7
CT25630	0.4	0.0	1.0	0.2	0.1	0.0	1.0	0.3	0.7	1.0	0.8	1.0	0.0	0.5	0.5	0.8	0.1	1.0
CT28453	0.8	0.7	0.9	0.6	1.0	0.8	0.6	0.9	1.0	1.0	0.6	0.8	0.7	1.0	1.0	1.0	0.7	0.6
CT35090	0.6	0.0	0.6	0.9	0.8	1.0	0.4	0.3	0.9	0.7	0.6	0.0	0.4	0.1	0.6	0.9	1.0	0.9
CT38500	0.3	0.7	0.9	1.0	0.5	0.4	0.5	0.8	0.0	1.0	1.0	0.1	0.0	0.5	0.3	0.0	0.4	0.6
CT42200	0.7	0.6	0.9	1.0	0.6	1.0	0.1	0.4	0.5	0.0	0.0	0.7	0.3	0.5	0.1	0.8	0.1	0.2
CT44256	0.0	1.0	0.8	0.7	1.0	1.0	0.6	0.1	0.7	0.9	0.5	1.0	0.7	0.9	1.0	0.3	0.2	0.0
CT46940	1.0	1.0	0.5	0.8	1.0	0.7	0.4	0.2	0.0	1.0	0.3	0.3	0.8	0.0	0.0	0.0	0.0	0.7
CT49600	1.0	1.0	0.0	0.0	0.0	0.5	0.9	0.7	1.0	1.0	1.0	0.4	0.7	1.0	0.5	0.6	0.0	0.2
CT57400	0.0	0.4	0.8	0.3	0.0	0.1	1.0	0.0	0.0	0.2	0.5	0.2	0.5	0.3	0.1	1.0	0.0	0.5
CT59060	0.5	1.0	0.8	0.0	0.0	0.1	0.9	0.8	1.0	1.0	1.0	0.5	0.9	1.0	0.8	1.0	1.0	0.5
CT65438	1.0	0.5	0.5	1.0	0.8	0.9	0.2	1.0	1.0	1.0	0.8	0.7	0.9	0.8	0.9	1.0	0.5	0.9
CT67760	1.0	0.3	0.5	0.5	0.9	0.4	0.4	0.1	0.0	0.3	0.7	0.6	1.0	0.9	0.9	0.3	0.9	0.8
CT73833	1.0	0.9	1.0	0.8	0.9	1.0	1.0	0.4	0.9	0.8	0.9	0.3	0.4	0.0	0.5	0.0	0.5	0.7
CT77820	0.4	0.3	0.0	1.0	0.9	0.2	1.0	0.0	0.8	1.0	0.9	0.6	0.0	0.4	0.1	0.4	0.0	0.0
CT78334	0.9	0.0	1.0	0.7	0.8	0.8	0.8	0.7	0.8	0.9	1.0	1.0	0.5	0.7	0.9	1.0	1.0	0.8
CT78990	0.6	0.8	1.0	1.0	0.9	0.6	0.5	1.0	1.0	0.3	0.4	0.1	0.0	0.4	0.0	0.6	1.0	0.9

CT79903	0.8	0.7	0.9	1.0	0.8	1.0	0.9	0.5	0.0	1.0	1.0	0.0	0.6	1.0	0.8	0.4	0.0	0.3
CT82840	0.5	0.9	0.1	0.0	0.7	1.0	0.0	1.0	1.0	1.0	1.0	0.5	1.0	0.9	1.0	0.3	0.1	0.4
CT84000	0.8	0.6	0.4	0.7	1.0	0.1	0.3	0.7	1.0	0.1	0.3	0.9	1.0	0.6	0.7	0.2	0.4	0.1
CT84676	0.3	0.4	0.1	0.4	0.5	0.2	0.1	0.3	0.8	1.0	0.8	1.0	0.8	0.3	0.6	0.2	0.2	0.7
CT85600	0.0	1.0	0.7	1.0	1.0	0.9	0.5	1.0	1.0	0.7	0.6	0.6	1.0	0.4	0.7	0.3	0.4	0.1
CT85634	1.0	0.1	0.7	0.0	0.0	0.3	0.7	0.5	0.5	0.0	0.2	0.5	0.0	1.0	1.0	0.3	0.1	0.7
CT85670	0.9	0.2	0.7	0.0	0.2	0.0	0.3	0.7	0.9	0.1	0.9	0.7	0.9	0.7	1.0	0.5	0.8	0.6
CT87490	1.0	1.0	0.5	0.9	0.6	0.7	0.3	0.5	0.0	0.5	0.1	0.2	1.0	1.0	0.7	0.9	0.7	1.0
CT89300	0.2	0.8	0.0	0.8	0.0	0.3	0.5	0.2	0.8	1.0	0.0	0.9	1.0	0.1	1.0	0.9	1.0	0.7
CT90092	0.8	0.0	0.3	0.1	0.1	0.0	0.0	0.2	0.0	0.5	0.4	0.8	0.3	0.9	0.8	0.1	0.4	0.0
CT90800	0.7	0.6	0.5	0.0	0.1	0.5	0.9	0.8	1.0	0.4	0.6	0.0	1.0	0.0	0.4	0.2	0.3	1.0
CT94500	0.2	1.0	1.0	0.8	0.1	0.0	0.8	0.5	0.1	0.3	1.0	1.0	0.8	0.3	0.4	0.7	0.6	0.7
CT94679	0.6	0.0	0.3	0.1	1.0	0.4	0.8	0.5	0.3	0.7	0.6	0.2	0.4	0.1	0.2	0.8	0.0	0.1
CT95700	0.7	0.8	0.9	1.0	1.0	1.0	1.0	0.7	0.8	1.0	1.0	0.5	1.0	1.0	1.0	0.2	0.0	0.2
CT98600	0.9	0.5	0.6	0.0	0.8	1.0	0.0	0.5	1.0	0.1	0.2	0.7	0.2	1.0	1.0	0.9	0.8	1.0

Survey Form:

Participants Information	
Name	
Qualification	
Designation	
Organization	
Contact No	

Q/A Area					
S-No	**Questions**	**Answers**			
General Selection Related Questions					
1.	Do you follow any particular approach for product's selection? State name and reason.	Yes	No	Name/Reason:	
2.	In-case of ad-hoc selection, what criteria do you use?				
3.	In your opinion, which task of selection process is most problematic? Why?				
4.	Would you prefer a proper method over ad-hoc selection? Keeping in mind that it would consume more time. State reason?	Yes	No	Reason:	
5.	Do you agree that following a proper method would increase chances of project's success?	Yes	No	Other:	
TAES-COTS Related Questions					
6.	Do you agree that this approach addresses all aspects of COTS product selection?	Yes	No	Other:	
7.	Do you agree that multiple criteria used in this approach will completely evaluate a product?	Yes	No	Other:	
8.	What in your opinion will be the effort required to implement this approach?	Low		Medium	High
9.	Do you agree that evaluators will not face much difficulty using this approach? If no? State area where you think they will face problems.	Yes	No	Comment:	
10.	Do you agree that the hierarchy of evaluation criteria reduces	Yes	No	Other:	

	products in every following step hence reducing the effort and increasing the implement-ability?			
11.	Do you agree that the final product suggested by this approach will be the optimal candidate?	Yes	No	Other:
12.	Do you agree that this approach is not time consuming, considering the results that will be generated using it?	Yes	No	Other:
13.	Do you agree that the techniques suggested for requirement gathering will help in collecting complete user requirements?	Yes	No	Other:
14.	Would you recommend using this approach in cases where huge numbers of products are to be evaluated?	Yes	No	Other:
15.	Would you prefer implementing this approach over others you followed? Why?	Yes	No	Other:
TAES-COTS's Detailed Evaluation Related Questions				
16.	Do you agree that the distribution of attributes into classes and then comparison of classes based on rules reduce the effort?	Yes	No	Other
17.	Do you think that the 81 rules are accurate enough to select best software candidate?	Yes	No	Other:
18.	In your opinion, what amount of time do you think will be needed to complete the evaluation?	Low		Medium / High
19.	Do you agree that COTSEVAL makes it simple to carry out the detailed evaluation?	Yes	No	Other:
20.	Do you believe that the 18 attributes are sufficient and they completely address the NFRs of any type of software?	Yes	No	Other:
21.	Do you agree that priorities in attributes are essential and increase the practicality?	Yes	No	Other: