ISSN 1512-1232

PATHWAYS TO EFFECTIVE SOFTWARE PRODUCT ENGINEERING: A COMPREHENSIVE AND STRATEGIC APPROACH

Dr.S.S.Riaz Ahamed

Principal, Sathak Institute of Technology, Ramanathapuram,India.
Email:ssriaz@ieee.org, ssriaz@yahoo.com

Abstract

Software engineering occurs as a consequence of a process called system engineering. Instead of concentrating solely on a software, systems engineering focuses on a business enterprise. When a product is to be built, it is called product engineering. Both information engineering and product engineering attempt to bring order to the development of computer-based systems. Although each is applied in a different application domain, both strive to put software into context. That is both information engineering and product engineering work to allocate a role for computer software and to establish the links that tie software to other elements of a computer based system.

Keywords: *Data Flow Diagram (DFD),*

1 INTRODUCTION

The goal of product engineering is to translate the customer's desire for a set of defined capabilities into a working product. To achieve this goal, product engineering like information engineering-must derive architecture and infrastructure.
The requirements for the development of a product may include the following:

- Software: Computer programs, data structures, and related documentation that serve to effect the logical method, procedure, or control that is required.
- Hardware: Electronic devices that provide computing capability, and electrochemical devices (e.g. sensors, motors, pumps) that provide external world function.
- People: Users and operators of hardware and software.
- Database: A large, organized collection of information that is accessed via software.
- Documentation: Manuals, forms, and other descriptive information that portrays the use/and or operation of the system.
- Procedure: The steps that define the specific use of each system element or the procedural context in which the system resides.

2 UNDERSTANDING THE CUSTOMER PROBLEM TO SOLVE:

2.1 Initial Investigation

The first step in the system development life cycle is the identification of a need. This is a user's request to change, improve, or enhance an existing system. Because there is likely to be a stream of such requests, standard procedures must be established to deal with them. The initial investigation is one way of handling this. The objective is to determine whether the request is valid and feasible before a recommendation is reached to do nothing, improve or modify the existing system, or build a new one.

The user's request form specifies the following

- User-assigned title of work requested.
- Nature of work requested (problem definition)

ISSN 1512-1232

- Data request was submitted.
- Data job should be completed.
- Job objective(s)--purpose of job requested.
- Expected benefits to be derived from proposed change.
- Input/output description--quantity (number of copies or pages) and frequency (daily, weekly, etc.) of inputs and outputs of proposed change.
- Requester's signature, title department, and phone number.
- Signature, title, department, and phone number of person approving the request.

The user request identifies the need for change and authorizes the initial investigation. There are three strategies for eliciting information regarding the user's requirements: asking questions, obtaining information from the present system, and prototyping. The asking strategy assumes a stable system where the user is well informed about information requirements. In contrast, the prototyping strategy is appropriate for high-uncertainty information requirements determination. It may undergo several modifications before it becomes a written commitment. Once the request is approved, the following activities are carried out: background investigation, fact-finding and analysis, and presentation of results--called project proposal. The proposal, when approved, initiates detailed user-oriented specification of system performance and analysis of feasibility of candidate system.

Fact-finding is the first step in the initial investigation. It includes a review of written documents, on-site observations, interviews, and questionnaires. The next step is fact analysis, which evaluates the elements related to the inputs and outputs of a given system. Data flow diagrams and other charts are prepared during this stage.

The data flow diagram (DFD) shows the flow of data, the processes, and the areas where they are stored. It is a commonly used structured tool for displaying the logical aspects of the system under study. Decision tables are used as a supplement when complex decision logic cannot tables are used as a supplement when complex decision logic cannot be represented clearly in a DFD.

The outcome of the initial investigation is to determine whether an alternative system is feasible. The proposal details the findings on the investigation. Approval of the document initiates a feasibility study, which leads to the selection of the best candidate systems.

Need Identification

The success of a system depends largely on how accurately a problem is defined, thoroughly investigated, and properly carried out through the choice of solution. User need identification and analysis are concerned with what the user needs rather than what the user wants. Not until the problem has been identified, defined, and evaluated should the analyst think of solutions and whether the problem is worth solving. This step is intended to help the user and analyst understand the real problem rather than its symptoms.

Other problems come into focus after a joint meeting between the user and the analyst. In either case, the user initiates an investigation by filling out a request form for information. The request provides for statements of objectives and expected benefits. The objectives of the problem situation must be understood within the framework of the organization’s BIS objective. If objectives are misunderstood, it is easy to solve the wrong problem. To successfully design the system, it requires a clear knowledge of what the system s intended to do.

Information Requirement

Shared, complete, and accurate information requirements are essential in building computer based information systems. Unfortunately, determining the information from each user needs is particularly difficult task. In fact, it is recognized as one of the most difficult tasks in system development. The system analyst determines the needs of the user and the information flow that will satisfy those needs. The usual approach is to ask the user what information is currently

ISSN 1512-1232

available and what other information is required. Interaction between the analyst and the user usually leads to an agreement about what information will be provided by the candidate system.

The user and analyst traditionally do not share a common orientation towards the problem definition. For example, in the analyst's view, the problem definition must be translatable into a system design expressed quantitatively in terms of outputs, inputs, processes, and data structures. This is an ideal way to develop a good system when all features are known, under the best of situations, and within time constraints. In contrast, the user seems to be satisfied with a qualitative definition that specifies the system in generalities. Flexibility is a key consideration. System specifications must change with their needs, as must the system after implementation. Based on this contrasting view, users who try to define their information requirements with the analyst's views find themselves in a predicament.

2.2 Strategies for Determining Information Requirements

There are three kinds of strategies or general approaches for eliciting information regarding the user's requirements:

1. Asking
2. Getting information from existing information system
3. Prototyping

Asking

This strategy obtains information from users by simply asking them about the requirements. It assumes a stable system where users are well informed and can overcome biases in defining their problem. There are three key asking methods.

- *Questions* may be open-ended or closed. An open-ended question allows the respondent to formulate a response. It is used when feeling or opinions are important. In contrast, a closed question requests one answer from a specific set of responses. It is used when factual responses are known.
- *Brainstorming* is a technique used for generating new ideas and obtaining general information requirements. This method is appropriate for eliciting no conventional solutions to problems. A guided approach to brainstorming is asking each participant to define ideal solutions and then select the best feasible one. It works well for users who have system knowledge by have difficulty accepting new ideas.
- *Group consensus* asks participants for their expectations regarding specific variables. For example, each participant fills out a questionnaire. The results are summarized and given to participants along with a follow-up questionnaire. Participants are invited to change their responses. The results are again summarized and fed back to the participants. This debate by questionnaire continues until participant's responses have converged enough. This method ha an advantage over brainstorming in that participants are not subjected to psychological pressure from other with presumed authority or influence.

Getting Information from the Existing Information system

Determining information from an existing application has been called the data analysis approach. It simply asks the user what information is currently received and what other information is required. It relies heavily on the user to articulate information needs. The analyst examines all reports, discusses with the user each piece of information examined, and determines unfulfilled information needs by interviewing the user. The analyst is primarily involved in improving the existing flow of data to the user.

ISSN 1512-1232

Prototyping

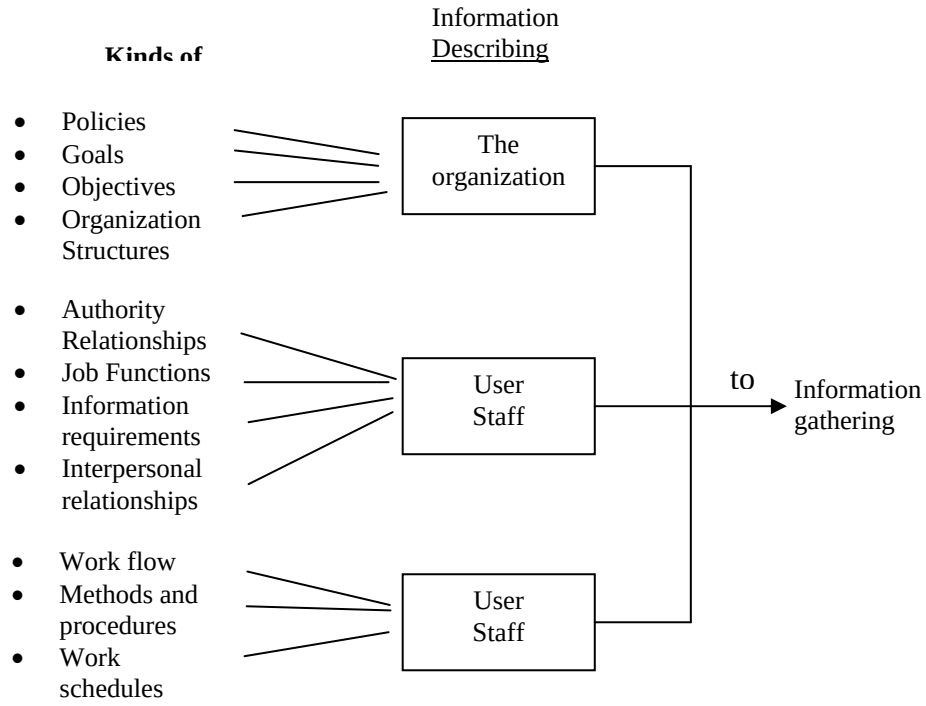
The third strategy for determining user information requirements is used when the user cannot establish information needs accurately before the information system is built. The reason could be the lack of an existing model on which to base requirement or a difficulty in visualizing candidate systems. In this case, the user needs to anchor on real-life systems from which adjustments can be made. Therefore, the iterative discovery approach captures an initial set of information requirements and builds a system to meet these requirements. As users gain experience in its use, they request additional requirements or modifications (iterations), in the system. In essence, information requirement are discovered by using the system. Prototyping is suitable in environments where it is difficult to formulate a concrete model for defining information requirements and where the information needs of the user are evolving, such as in DSS.

2.3 Information Gathering

Whether the thrust of the activities is the initial investigation or a feasibility study, the aim is primarily to develop an understanding of the problem facing the user and the nature of operation. Understanding how each activity operates requires access to information.

Information gathering is an art and science. The approach and manner in which information is gathered require persons with sensitivity, common sense, and knowledge of what and when to gather and what channels to use in securing information. Additionally, the methodology and tools for information gathering require training and experience that the analyst is expected to have. This means that information gathering is neither easy nor routine. Much operation, experience and training are required.

Categories of Information



Categories of Information

Before one determines where to go for information or what tools to use, the first requirement is to figure out what information to gather. Much of the information needed to analyze relates to the organization in general, the user staff, and the workflow as show in the above figure.

ISSN 1512-1232

Information about the organization

Information about the organization's policies, goals, objectives, and structure explains the kind of environment that promotes (or hinders) the introduction of computer-based systems. Company policies are guidelines that determine the conduct of business. Policies are translated into rules and procedures for achieving goals. A statement of goals describes management's commitment to objectives and the direction system development will follow. Objectives are milestones of accomplishments towards achieving goals. Information from employee manuals, orientation pamphlets, and annual company reports, and the like helps an analyst form opinions about the goals of organization.

Information about User Staff

Another kind of information for analysis is knowledge about the people who run the present system – their job functions and information requirements, the relationships of their jobs to the existing system, and the interpersonal network that holds the user group together. The major focus is to find out what people the analyst is going to be dealing with and what each person expects to get out of a candidate system before it goes through design and final implementation. Once such information is secured, the next step is to show how various jobs hang together within work schedules and procedures.

Information about Workflow

Workflow focuses on what happens to the data through various points in a system. This can be shown by a data flow diagram or a flowchart. A data flow diagram represents the information generated at each processing point in the system and the direction it takes from source to destination. In contrast, a system flowchart describes the physical system. The information available from such charts explains the procedures used for performing tasks and work schedules.

Information Gathering Tools

No two projects are ever the same. This means that the analyst must decide on the information gathering tool and how it must be used. Although there are no standard rules for specifying their use, an important rule is that information must be acquired accurately, methodically, under the right conditions, and with minimum interruption to user personal. The information's can be gathered in four different ways:

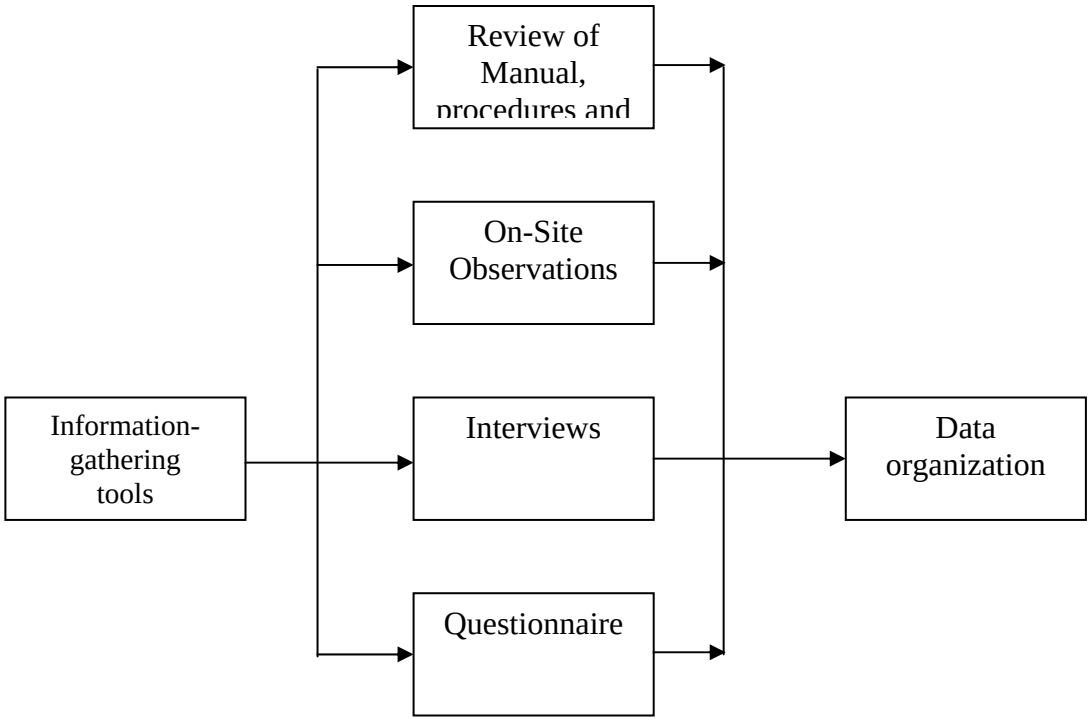
- Information available in existing manuals
- On-site observations
- Interviews
- Questionnaires

Each tool has a special function, depending on the information needed.

Review of Manuals, procedure, and Forms

Very few system problems are unique. The increasing number of software packages suggests that problem solutions are becoming standardized. Procedures manuals and forms are useful sources for the analyst. They describe the format and functions of the present system. Included in most manuals are system requirements that help determine how well various objectives are met. Up-to-date manuals save hours of information gathering time. Unfortunately, in many cases, manuals do not exist or are seriously out of date.

ISSN 1512-1232



Information Gathering Methods

On-site Observation

Another information gathering tool used in system studies is on-site observation. It is the process of recognizing and noting people, objects and occurrences to obtain information. The analyst role is that of an information seeker who is expected to be detached (therefore unbiased) from the system being observed. This role permits participation with the user staff openly and freely.

The major objective of on-site observation is to get as close as possible to the 'real' system being studied. For this reason it is important that the analyst is knowledgeable about the general makeup and activities of the system. The following questions serve as a guide for on-site observations:

1. What kind of system is it? What does it do?
2. Who runs the system? Who are the most important people in it?
3. What is the history of the system? How did it get to its present stage of development?
4. Apart from its formal function, what kind of system is it in comparison with other systems in the organization? Is it a primary or a secondary contributor to the organization? Is it fast paced or is it a leisurely system that responds slowly to external crises?

Four alternative observations that can be considered are

1. Natural or contrived: A natural observation occurs in setting such as employee's place of work and a contrived observation is setup by the observer in a place like a laboratory.
2. Obtrusive or unobtrusive: An obtrusive observation takes place when the responded knows of being observed, and an unobtrusive takes place in a contrived way such as behind a one-way mirror.
3. Direct or indirect: A direct observation takes place at the work spot when at work and indirect observation uses mechanical devices such as camera to capture information.

ISSN 1512-1232

4. Structured or Unstructured: The structured observation takes place in an structure format that how work carries out and in unstructured format, the observation will not be in structured but later has to be ordered as the workflow.

Interviews and Questionnaires

Information gathering tools can be categorized by their degree of directness. If we wish to know about something, we simple ask someone about it directly, but we may not get an answer. Most of the information gathering tools used in system analysis are relatively direct. This is the strength because much of the information needed can be acquired by direct questions.

Interviews is a face-to-face interpersonal role situation in which a person called the interviewer asks a person being interviewed questions designed to gather information about a problem area. The interview is the oldest and most often used device for gathering information in systems work. It has qualities that behavioral and on-site observations do not posses. It can be used for two main purposes:

- As an exploratory device to identify relations or verify information,
- To capture information as it exists

Validity is not a small problem; Special pains are taken to eliminate interviews bias. It is assumed that information is more valid, the more freely it is given. Such an assumption stresses the voluntary character of the interview as a relationship freely and willingly entered into by the respondent. If the interview is considered a requirement, the interviewer might gain the respondent's time and attention, but cannot be certain of the accuracy of the information gathered during the interview.

In an interview, since the analyst (interviewer) and the person interviewed meet face to face, there is an opportunity for flexibility in eliciting information. The analyst is also in a position to observe the subject.

There are four primary advantages of the interviewer:

- Its flexibility makes the interview a superior technique for exploring areas where not much is known about what questions to ask or how to formulate questions.
- It offers a better opportunity than the questionnaire to evaluate the validity of the information gathered. The interviewer can observe not only what subjects say but also how they say it.
- It is an effective technique for eliciting information about complex subjects and for probing the sentiments underlying expressed opinions.
- Many people enjoy being interviewed, regardless of the subject. They usually cooperate in a study when all they have to do is talk. In contrast, the percentage of returns to a questionnaire is relatively low: often less than 20 percent. Attractively designed questionnaires that are simple to return, easy to follow, and presented in a context that inspires cooperation improve the return rate.

Sometimes an old-fashioned interview is the best way to find out exactly what the customer wants, especially if you don't fully understand what the current situation is and what the goal is. Use questions like these to probe for your customer's needs:

- What is working for you now?

ISSN 1512-1232

- What parts of the current system would you most like to replace?
- Are there additional reports you would like to be able to generate?
- What items would you most like to keep track of?
- Is the data private or public?
- Who needs access to the data and what kind of access should each user or group have?
- Would you like the data posted on the Internet?
- Would you like the public to be able to look up things via the Internet?
- Do you have sufficient hardware in place to run both the database server and client software?
- If money and technology were no object, what would you like to incorporate into the new system?

By asking these kinds of questions you can quickly build a sense of why a database is needed. Although you might not be able to provide everything (given the limitations of the assigned budget, time frame, and hardware allowances), you will have the start of a long-term plan for growth and expansion of the database.

The major drawback of the interview is the long preparation time. Interviews also take a lot of time to conduct, which means time and money. So whenever a more economical alternative captures the same information, the interview is generally not used.

Questionnaire is a term used for almost any tool that has questions to which individuals respond. It is usually associated with self-administered tools with items of the closed or fixed alternative type. By its nature, a questionnaire offers the following advantages:

- It is economical and requires less skill to administer than the interview.
- Unlike the interview, which generally questions one subject at a time, a questionnaire can be administered to large number of individuals simultaneously.
- The standardized wording and order of the questions and the standardized instructions for reporting responses ensure uniformity of questions. In contrast, the interview situation is rarely uniform from one interview to the next.
- The respondents fell greater confidence in the anonymity of a questionnaire than in that of an interview. In an interview, the analyst usually knows that user staff by name, job function, or other identification. With a questionnaire, respondents give opinions without fear that the answer will be connected to their names.
- The questionnaire places les pressure on subjects for immediate responses. Respondents have time to think the questions over and do calculation to provide more accurate data.

The advantage of the self-administered questionnaire outweighs disadvantages, especially when cost is consideration. The principal disadvantage is a low percentage of returns. Another disadvantage is that many people have difficulty in expressing themselves in writing, especially when responding to essay (open) questions. Many dislike writing. Because of these disadvantages, the interview is probably superior to the questionnaire.

ISSN 1512-1232

3 PRODUCT OBJECTIVES

The first step in designing any project is to define the problem. It is difficult to answer a question if you don't know which question is being asked! The Problem definition involves in making some decisions about the direction and goals of the Project. But it will not involve making specific design decisions. At the end of this process, the team should be able to state the goal of the project in a few sentences. These sentences will be known as the objectives. In some industrial settings, these are also known as the deliverables. It is required to be very careful when choosing the objectives. If they are too complex, it will be very difficult to satisfy them, but if they are too simple, then the project will not demonstrate the full capabilities of the group. Once written the project objectives, the criteria for each of the objectives are selected. Criteria are ways in which you can tell if you met your objectives. They should be easily measurable, so someone not involved in the project will be able to tell if you met your objectives or not, just by seeing a simple demonstration.

Three types of joint management reviews are conduced throughout the product development process:

1. Major Milestones: These systemwide events are held at the end of each development phase. They provide visibility to systemwide issues, synchronize the management and engineering perspectives, and verify the aims of the phase have been achieved.
2. Minor milestones: These iteration-focused events are conducted to review the content of an iteration in details and to authorize continued work.
3. Status assessments: These periodic events provide management with frequent and regular insight into the progress being made.

Each of the four phases-inception, elaboration, construction, and transition- consists of one or more iterations and concludes with major milestone when a planned technical capability is produced in demonstrable form. An iteration represents a cycle of activities for which there is a well-defined intermediate result-a minor milestone-captured with two artifacts: a release specification (the evaluation criteria and plan) and a release description (the results). Major milestones at the end of each phase use formal, stakeholder-approved evaluation criteria and release descriptions; minor milestones use informal, development-team-controlled versions of these artifacts. The level of ceremony and the number of milestones will vary depending on several parameters, such as scale, number of stakeholders, business context, technical risk, and sensitivity of cost and schedule perturbations. Different stakeholders have different concerns:

- Customers: schedule and budget estimates, feasibility, risk assessment, requirements, understanding, progress, product line compatibility.
- Users: consistency with requirements and usage scenarios, potential for accommodating growth, quality attributes
- Architects and systems engineers: product line compatibility, requirements changes, trade-off analyses, completeness and consistency, balance among risk, quality and usability.
- Developers: sufficiency of requirements details and usage scenario descriptions, frameworks for component selection or development, resolution of development risk, product line compatibility, sufficiency of the development environment.
- Maintainers: sufficiency of product and documentation artifacts, understandability, interoperability with existing systems, sufficiency of maintenance environment.

ISSN 1512-1232

- Others: possibly many other perspectives by stakeholders such as regulatory agencies, independent verification and validation contractors, venture capital investors, subcontractors, associate contractors, and sales and marketing teams.

The general status of plans, requirements, and products across the major milestones are described in this table.

MILESTONES	PLANS	UNDERSTANDING OF PROBLEM SPACE (REQUIREMENTS)	SOLUTION SPACE PROGRESS(SOFTWARE PRODUCT)
Life-cycle objectives milestones	Definition of stakeholder responsibilities Low-fidelity-life-cycle plan High-fidelity elaborate phase plan	Baseline vision, including growth vectors, quality attributes, and priorities Use case model	Demonstration of atleast one feasible architecture Make/buy/reuse trade-offs Initial design model
Life-cycle architecture milestone	High-fidelity construction phase plan (bill of materials, labor allocation) Low-fidelity transition phase plan	Stable vision and use case model Evaluation criteria for construction releases, initial operational capability Draft user manual	Stable design set Make/buy/reuse decisions Critical component prototypes
Initial operational capability milestone	High-fidelity transition phase plan	Acceptance criteria for product release Releasable user manual	Stable implementation set. Critical features and core capabilities Objective insight into product qualities.
Product release milestone	Next-generation product plan	Final user manual	Stable deployment set Full features Compliant quality

Default content of status assessment reviews

TOPIC	CONTENT
Personnel	Staffing plan vs. actuals Attrition, additions
Financial trends	Expenditure plan vs. actuals for the previous, current, and next major milestones Revenue forecasts
Top 10 risks	Issues and criticality resolution plans Quantification (cost, time, quality) of exposure
Technical progress	Configuration baseline schedules for major milestones Software management metrics and indicators Current change trends Test and quality assessments
Major milestone plans and results	Plan, schedule, and risks for the next major milestone pass/fail results for all acceptance criteria.
Total product scope	Total size, growth, and acceptance criteria perturbations.

ISSN 1512-1232

In general the objectives of the product are:

- ***Maintainability***
  - It should be possible for the software to evolve to meet changing requirements
- ***Dependability***
  - The software should not cause physical or economic damage in the event of failure
- ***Efficiency***
  - The software should not make wasteful use of system resources
- ***Usability***
  - Software should have an appropriate user interface and documentation

4 CONCLUSION

A new product emerges when the product in existence does not satisfy the need for which the product is intended. Every product has its own inherent limitations and usage. This gives room for the development of the new product either by modification of the already existing product or by translation of a new idea, which takes the form of a new product.

5 REFERENCES

1. David I. Cleland, Roland Gareis (2006). Global project management handbook. McGraw-Hill Professional, 2006. ISBN 0071460454. Pp.1-4.
2. Martin Stevens (2002). Project Management Pathways. Association for Project Management. APM Publishing Limited, 2002.
3. Morgen Witzel (2003). Fifty key figures in management. Routledge, 2003. ISBN 0415369770. Pp. 96-101.
4. Bjarne Kousholt (2007). Project Management –. Theory and practice.. Nyt Teknisk Forlag. ISBN 8757126038. p.59.
5. F. L. Harrison, Dennis Lock (2004). Advanced project management: a structured approach. Gower Publishing, Ltd., 2004. ISBN 0566078228. p.34.
6. Stellman, Andrew; Greene, Jennifer (2005). Applied Software Project Management. O'Reilly Media. ISBN 978-0-596-00948-9.
7. Albert Hamilton (2004). Handbook of Project Management Procedures. TTL Publishing, Ltd. ISBN 07277-3258-7.
8. Roger S. Pressman, Software engineering : A practitioner's approach, MCGraw-Hill publication, 1997, ed.4
9. Edward Kit, Software testing in the real world, Addison-Wesley publications, 2000, ed.1
10. Ian Sommerville, Software engineering, Addison-Wesley publications, 1996, ed.5
11. Pankaj Jalote, An integrated approach to software engineering, Narosa publications, 1997, ed. 2
12. Shari Lawrence Peleeger, software engineering theory and practice, Pearson education, 2001, ed. 2
13. Richard Fairly, Software engineering concepts, McGraw-Hill Inc.,1985
14. Myers, Glenford J. (1979). The Art of Software Testing. John Wiley and Sons. p. 145-146. ISBN 0-471-04328-1.
15. Barry W. Boehm, Software Engineering Economics, Prentice-Hall Inc., 1981.
16. Dustin, Elfriede (2002). Effective Software Testing. Addison Wesley. p.3. ISBN 0-20179-429-2.
17. Carlo Ghezzi et al : Fundamentals of Software Engineering (Fifth edition) PHI 1991.

Article received: 2010-04-24