

# **A Qualitative Study on Educational Software Requirements Analysis – The Analysis Phase of SDLC**

**B. Stiram**

Quality Assurance Officer, Lecturer, IST  
Department  
Sur University College  
Sur, Sultanate of Oman

**S. Bhuvana**

Lecturer, IT Department  
JRS College  
Scarborough, Ontario M1S 5H6, Canada

## **ABSTRACT**

Technological development introduced various contemporary teaching and learning tools in educational processes. The educational processes have significant impacts because of the new educational technology. Educational technology enhances the teaching and learning processes with the available tools and techniques. Various contextual and conceptual developments have taken place in educational environment. Though the educational technologies have both positive and negative impacts on educators and learners, the technological usage is increasing in daily activities. Educational software have prominent role in educational activities. A proper development methodology has to be followed while such educational software is developed. The Software Development Life Cycle (SDLC) defines the various phases of software development. The first phase of SDLC is requirement analysis.

The primary goal of this study is to identify the software requirements in various perspectives. This paper focuses on analyzing the software requirements with respect to the end users such as educators, learners and management. The paper suggests a requirement analysis influence factors that have to be considered and studied thoroughly while developing educational software.

## **Keywords**

educational software, Software Development Life Cycle (SDLC), software requirements, requirement analysis framework, teaching and learning processes

## **1. INTRODUCTION**

### **1.1. General Introduction**

Technological development in this modern era has changed the traditional educational processes into various contemporary processes. Technology provides different modern tools and techniques that enhance the educational environment into various degrees. Technological innovations play a significant role in revolutionizing educational processes with respect to educators and learners perceptions.

Information technology provides numerous supports in day – to – day teaching and learning processes. The contemporary educational processes use these information technology and systems in conjunction with traditional processes. The educational software is used in various aspects of the teaching and learning processes to enhance the educational activities. Educational software's are developed with various perspectives to meet the educators and learners perceptions. Apart from this,

the institution, that adopts such educational software, perceptions have to be considered while developing the educational software.

The Software Development Life Cycle (SDLC) provides different models and phases in developing such educational software. The models proposed in software engineering and systems analysis and design, define different approaches in software development. Some SDLC models are The Waterfall Model, Evolutionary Development, Component – Based Software Engineering, Generic Software Process Model, Spiral Model, Iterative and Incremental Development and Agile Development Model.

Though the above said models define different phases for SDLC, the common phases are Planning, Requirement Analysis, Design, Implement and Validate, and Maintenance. The educational software development should follow the SDLC in conjunction with technological innovations and potentials.

Our study focuses on the “Requirement Analysis” phase of SDLC in educational software. The primary goal of this study is to identify the software requirements in various perspectives. This paper focuses on analyzing the software requirements with respect to the end users such as educators, learners and management. The paper suggests a requirement analysis framework that has to be considered and studied thoroughly while developing educational software.

### **1.2. Statement of Problem**

Though the currently available educational software's serve the educational processes in some aspects, still there are some discrepancies between the technology potentials and educational processes expectations. Some educational software's lack in various aspects of educators, students and management perspectives. The educational software development should include all these requirements and expectations to assist and improve educational processes.

### **1.3. Aims and Objectives**

The primary goal of this study is to identify the educational software requirements with respect to various users such as educators, learners and management. Also, the major objective of this study is define a common requirement framework that may be followed while developing educational software to optimize the software usage in educational processes.

### **1.4. Importance of Study**

The basic idea of this research is to assist the developers to identify the factors that have to be concentrated in educational

software development. A requirement analysis framework has been proposed to help the developers, educators, learners, and management to monitor the educational software development pertaining to teaching and learning processes. The researchers shall further divide the framework in various perspectives to optimize the technology potentials and applications in different educational processes.

### 1.5. Research Questions

- 1) Which major factors have to be analyzed while developing educational software?
- 2) How the educational needs shall be incorporated in educational software in order to optimize the usage in daily activities?
- 3) What are the dimensions to be considered in educational software development processes?

### 1.6. Some common definitions in the study

In this study, “management” refers to the people involved in buying and implementing the educational software in the educational environment.

## 2. LITERATURE REVIEW

Sommerville (2009) [14] has defined a software process has a set of activities that lead to the production of a software product. Software processes are complex and rely on people making decisions and judgements. A software process model is an abstract representation of a software process. The generic models of SDLC are not definitive descriptions of software processes. They may be extended adapted to create more specific software engineering processes. Figure 1 shows the waterfall model of software life cycle.

Waman (2010) [16], describes the activities involved in the SDLC cycle. He said that each process model has strengths and merits that make it suitable for specific applications. However, all processes are the same so far as some activities are concerned. These activities by default are

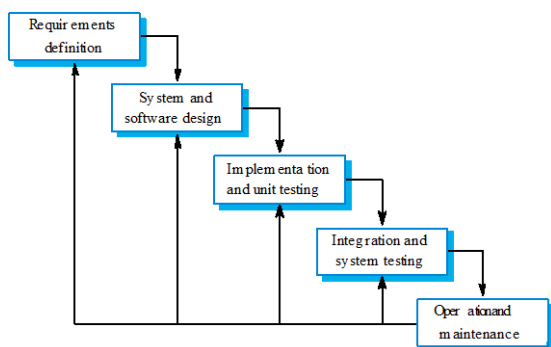


Fig 1: The Software Life Cycle

Source: Software Engineering (Sommerville, 2009)

Required to be carried out irrespective of the process. Some of the common activities are: planning, scheduling, tracking and control of activities. The task is made well defined precise work breakdown, each having a clear measurable goals of achievement, is known as Milestone. Requirement analysis and definition is stable and unlikely to change in a significant way during development and in the post implementation period. The requirement and scope get enlarged and improved in each iterative development process, making the system better in every respect.

Ronald (2001) [17], discussed different processes of SDLC. Lightweight methodologies on the other hand are a compromise between no process and too much process. These new methods were developed to efficiently manage software projects subjected to short timelines and excessive uncertainty and change. Nine types of lightweight SDLCs are Adaptive Software Development (ASD), Agile Software Process (ASP), Crystal, Dynamic System Development Method (DSDM), Extreme Programming (XP), Feature Driven Development (FDD), Rational Unified Process (RUP), SCRUM, and Whitewater Interactive System Development with Object Models (Wisdom). Strengths of these new light methodologies include their simpler processes and easier acceptance by developers who are only familiar with code and fix techniques. In addition, these lightweight SDLCs aid developers in thinking clearly about the end products they are creating. Disadvantages include their inability to handle large development teams. Lightweight methodologies are most appropriate when there are uncertain and volatile requirements, responsible and motivated developers, and customers who wish to become involved. On the other hand, lightweight methods are inappropriate with teams of more than fifty and/or the project has a fixed scope.

## 3. RESEARCH METHODOLOGY

### Data Collection:

To conduct this research a qualitative study was conducted. The educators and learners in Sur University College were interviewed for their respective requirements. Also, Sur University College administrative people were interviewed to know the management perspectives. Out of 80 students, 20 (25%) of the students were conducted face – to – face interview. Other students were asked to write their requirements for developing educational software.

Some available educational software’s were analyzed for the software potentials. Test managers distributed with some books were tested for their usage. In addition, researches conducted by various other authors were studied for getting clear ideas of requirements. Data collected for another research work “Educational Hypermedia Development Methodologies” (Salah and Sriram, 2010) [1], were analyzed partially to identify the educational software requirements.

Table 1 shows the number of learners interviewed for learner’s requirements.

During interview, nearly 80% of diploma students preferred traditional blackboard teaching in most of the cases. Nearly 60% of bachelor students preferred a combination of traditional and contemporary teaching and learning processes.

Table 2 shows the educator’s profile with their qualification. The language teachers were teaching English at foundation and diploma levels. The IT and Business instructors were teaching diploma and bachelor levels.

75% of the language teachers preferred the combination of traditional and modern teaching methods. They felt that the foundation level students were not used to any educational software. The learners at this level may feel difficult to understand the concepts and contents in contemporary teaching methodology using educational software. Nearly 66% of the other department staffs preferred educational software in their teaching processes. Nearly 33% of the educators suggested that the traditional blackboard teaching would be effective in quantitative subjects. Their software requirements were analyzed in both teaching and learning perspectives.

**Table 1: Number of Learners interviewed – Gender wise and Level Wise**

Gender	Diploma	Bachelor	Total
Male	10	9	19
Female	20	41	61
Total	30	50	80

**Table 2: Educator's Profile**

	Language	IT	Business	Total
Bachelor's	1	0	0	2
Master's	3	2	1	6
Ph.D's	0	1	2	2
Total	4	3	3	10

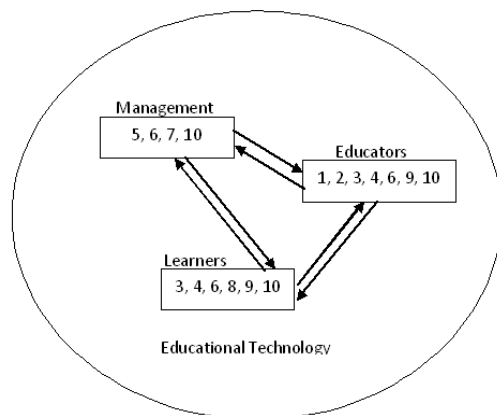
#### 4. FINDINGS

The various software requirements were studied and analyzed with respect to learners and educators and management perspectives. As the learners and educators are the major end users of the educational software, the analysis phase should consider their requirements with respect to technology potentials. The feasibility study, factors analysis and environmental analysis have to be conducted in order to identify the accurate educational software development.

The following major factors were identified in this study in learners, educators and management perspectives.

- a) Content Management
- b) Course Delivery
- c) Educational Technology Potentials
- d) Adaptation Models
- e) Infrastructure Requirements
- f) Resource Requirements
- g) Cost Requirements
- h) Students Motivation and Participation
- i) Assessment and Evaluation Tools
- j) Socio-economic Factors

Figure 2 shows the factors influences on various perspectives.



**Fig 2: Influencing Factors**

#### 5. ANALYZING EDUCATIONAL SOFTWARE REQUIREMENTS – ANALYSIS PHASE OF SDLC

The analysis phase of SDLC, identify the functional and non-functional requirements, user requirements, system requirements and interface specification. The functional requirement specifies the system behavior in any circumstances. The non-functional requirement specifies the constraints on the services. Domain requirement defines the characteristics of the application domain. The user requirements for a system have to be identified with all required conceptual information. The system requirements declare the user requirements in system design. The interface specification describes the precise interface that will suit to the current system (Soomerville, 2009)

##### 5.1. Analyzing the Software Requirements in Various Perspectives

###### 5.1.1. Management Perspectives

The major factors that are identified as per management perspectives are: Infrastructure Requirements, Cost Requirements, Resource Requirements and Socio-economic Factors. The analysis phase should make required feasibility study based on these factors in order to provide necessary assistance in these regards.

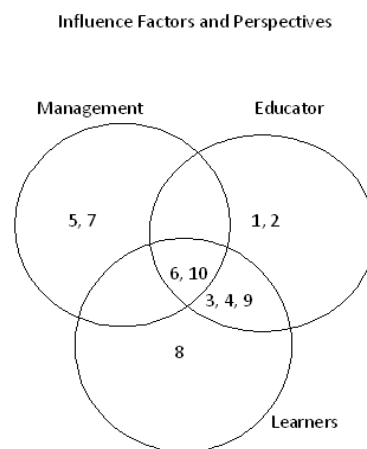
###### 5.1.2. Educator's Perspectives

Based on the analysis the optimum usage of the educational software by the educators depends on the following factors: Content Management, Course Delivery, Educational Technology Potentials, Adaptation Models, Assessment and Evaluation Tools and Socio-Economic Impact Factors.

###### 5.1.3. Student's Perspectives

As the students are one of the major end users of the educational software, their requirements need to be studied and analyzed with proper care. The major factors identified that influences the students are: Educational Technology Potentials, Adaptation Models, Resource Requirements, Students Motivation and Participation, Assessment and Evaluation Tools and Socio-Economic Factors.

Fig 3 describes the influence factors with their perspectives.



**Fig 3: Influence Factors and Perspectives**

## 5.2. Requirement Analysis

### 5.2.1 Content Management

The educational software should provide necessary tools and techniques for the corresponding course content management. Easy upload and use facilities should be provided as an additional tool. The course content should be covered to meet the required learning outcomes.

### 5.2.2 Course Delivery

As the teaching methods change in the heterogeneous educational environment, proper course delivery methods should be provided in the software. For the different courses, various course delivery tools should be included. Separate course delivery methods should be modeled for quantitative courses. Easy access to various sources and chapters should be provided.

### 5.2.3 Educational Technology Potentials

Educational technology has changed the teaching and learning methodologies drastically. Due to increasing innovations, teaching and learning methodologies have different degrees of dimensions. The proper teaching methodology needs to be analyzed as a prime factor.

### 5.2.4 Adaptation Models

The educators and learners cognitive psychologies have to be analyzed as adaptation model. The educational technology potentials and educational perceptions have to be analyzed rigorously. A proper adaptation models need to be specified with educators and learners cognitive skills impacts.

### 5.2.5 Infrastructure Requirements

The required infrastructures such as software, hardware, internet facilities, and other basic and advanced infrastructure requirements have to be mentioned clearly.

### 5.2.6 Resource Requirements

The resource requirements should specify the required books, journals, articles and other online and database requirements. These requirements should also classify the resource requirements for the educators and the learners.

### 5.2.7 Cost Requirements

The cost of the software should be within a permissible amount that is affordable by the management. The study should also specify the minimum and maximum cost requirements of the hardware, software, resources and other educational technology requirements.

### 5.2.8 Students Motivation and participation

The core concept of the educational software should concentrate on student's participation in motivation. As students' participation plays a vital role in educational processes, proper facilities should be provided with the educational software. Also, the educational software should motivate the students towards studies.

### 5.2.9 Assessment and Evaluation Tools

Various internal and external assessment and evaluation schemes have to be provided with the educational software. As assessment methods depend on each course, different assessment and evaluation techniques have to be developed. The educators should have enough permission to modify the assessment and evaluation methods.

### 5.2.10 Socio – economic Factors

These factors should analyze the social, cultural and economic impacts of the educational software in the specified environment.

## 6. CONCLUSION

The major factors influencing the usage of educational software in teaching and learning processes have to be analyzed and implemented while developing the educational software. Their impacts also have to be studied in various points of views before software development. The major factors identified in this research need to be incorporated in educational software in order to optimize the usage in daily activities. Software development team should consider the management, educators and learners perspectives for effectiveness.

## 7. FUTURE SCOPE

This qualitative study may be further extended into quantitative study to identify the influencing factors. The analysis on requirements may also further intensively studied for user requirements, system requirements and other requirements.

## 8. ACKNOWLEDGEMENT

We would like to thank the Sur University College, Sur for its research supports. Also, we would like to thank the educators, students and management people for their valuable inputs.

## 9. REFERENCES

- [1] AlKhafaji, Salah., Sriram, B., - "Educational Hypermedia Development Methodologies" – Presented in First Joint Scientific Symposium Committee, CAS, Sultanate of Oman on 13<sup>th</sup> April 2010 in Sohar University, Sultanate of Oman – Conference Proceedings PP: 8 – 24
- [2] Costa, A.P., Loureiro, M.J., Reis, L.P., - "Development Methodologies for Educational Software: The Practical Case of Courseware SERE" - International Conference on Education and New Learning Technologies (EDULEARN09) - 2009
- [3] Flemmer, Chip., - Technology's Influence on SCLEs – Visited and Retrieved on 16.03.2011 from <https://sites.google.com/a/boisestate.edu/edtech504/lflemmer>
- [4] Hoffer, J. A., George, J. F., Valacich, J.S., – Modern Systems Analysis and Design – 5e – (2009) - Pearson Education
- [5] Jalote, Pankaj – Software Engineering A Precise Approach – (2010) - Wiley India Private Ltd
- [6] KANUKA , HEATHER., ANDERSON, TERRY., - USING CONSTRUCTIVISM IN TECHNOLOGY-MEDIATED LEARNING: CONSTRUCTING ORDER OUT OF THE CHAOS IN THE LITERATURE - RADICAL PEDAGOGY - (1999) - ISSN: 1524-6345 VISITED AND RETRIEVED FROM [HTTP://RADICALPEDAGOGY.ICAAP.ORG/CONTENT/ISSUE1\\_2/02KANUKA1\\_2.HTML](http://radicalpedagogy.icaap.org/content/issue1_2/02KANUKA1_2.HTML) ON 16.03.2011
- [7] O'Sullivan, Mary Low., Samarawickrema, Gayani., - Changing learning and teaching relationships in the educational technology landscape - *Proceedings ascilite Melbourne 2008:-711 - 714*
- [8] Pape, Bernd., Bleek, W. G., Jackewitz, Iver ., Janneck, Michael., - Software Requirements for Project-Based Learning – CommSy as an Exemplary Approach - Proceedings of the 35th Hawaii International Conference on System Sciences - 2002

- [9] Peters, J. S., Pedrycz Witold – Software Engineering: An Engineering Approach – (2010) - Wiley India Pvt Ltd
- [10] Puustinen., M. W., Bakerz, M., & Lundz, K., - GESTALT: a framework for redesign of educational software - Journal of Computer Assisted Learning 22, - (2006) - Blackwell Publishing Ltd - pp34–46
- [11] Quintana, Chris., and Soloway, Elliot., Norris, Cathleen., - Learner-Centered Design: Developing Software that Scaffolds Learning - Proceedings of the IEEE International Conference on Advanced Learning Techniques (ICALT'01) - (2001) - 0-7695-1013-2/01
- [12] Shelly, G. B., Cashman, T. J., Rosenblatt, H. J. – Systems Analysis and Design – 4e – (2003) - Thomson Course Technology
- [13] Solomonidou, Christina., – “Constructivist design and evaluation of interactive educational software: a research-based approach and examples” - Open Education - The Journal for Open and Distance Education and Educational Technology - V5, N1, (2009) - Open Education ISSN: 1791-9312
- [14] Sommerville, Ian., – Software Engineering – 9e - (2009) – Pearson Education
- [15] Williamson, A., Kennedy, D. M., McNaught, C. and DeSouza, R.. Issues of intellectual capital and intellectual property in educational software development teams. *Australian Journal of Educational Technology*, 19(3), (2003) - 339-355.  
<http://www.ascilite.org.au/ajet/ajet19/williamson.html>
- [16] Waman, Jawadekar, S., – Software Engineering Principles and Practice – 12<sup>th</sup> Reprint – (2010) - Tata McGraw – Hill
- [17] Wolak, Ronald G., - Systems Development Research paper I – SDLC on a Diet – DISS 725 – spring 2001.