http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 5 (4): 726-734, 2006
ISSN 1812-5638

Systematic Multimedia Courseware Development Using a Software Engineering Style Process Model

T.S.K. Homeed and A. Mahmood
Department of Computer Science, University of Bahrain, Kingdom of Bahrain

Abstract: Multimedia and internet technologies have opened up new opportunities to be exploited in distance learning environment. These technologies have the ability to combine, integrate and deliver various media such as text, graphic, audio, video and animation which can help make teaching and learning more effective and rewarding. However, a high-quality and cost-effective multimedia courseware development is a major challenge that needs to be addressed if we want to make multimedia and internet-based distance learning a real success. This study presents a multimedia courseware engineering process model. The proposed model is an adaptation of the spiral model, a well-known model from software engineering field. Various phases and activities of the model are discussed in detail. We also present a methodology for courseware quality assurance and its relationship with the development process model. Our experience with developing a number of courseware using the proposed model is also presented.

Key words: Distance learning, multimedia courseware, e-learning, tutoring system, software engineering

INTRODUCTION

Distance education is an established method of teaching in which tutors and students are separated by physical distance. It has many benefits as compared to formal face-to-face education delivery system. For example, it provides a chance of learning to those who are deprived of formal education due to social, economical, geographical and other reasons. It also helps those who could not continue their education due to job commitments, limited time, non-availability of suitable educational or vocational programs, physical disability and other reasons. In addition, distance technical and vocational education provides opportunities to semi-skilled and unskilled workforce to enhance their qualifications and skills. Due to many advantages that the distance education offers, many universities around the world are now offering distance education programs.

In a traditional distance learning environments, the students are provided with course material (text books, video lectures, assignments/homework etc.), generally, through postal services. This is not only expensive but also an inefficient course delivery method. Also, the printed matter is not changed frequently due to the cost involved. However, recent developments in multimedia and Internet technologies have opened new opportunities to be exploited in distance education. It has been widely recognized that the use of multimedia and internet technologies would create a major shift in the educational service paradigm that promises major advantages over the analog distance learning and face-to-face systems (Carver *et al*., 1999; Lee and Sullivan, 1996). Radical changes in the computing infrastructure, spurred by multimedia computing and advanced communication technology will make the lecture theatres and laboratories much more accessible and effective (Carver and Biehler, 1994). While the multimedia and Internet provide many opportunities but it also poses many challenges to multimedia courseware developers. A common illusion in open and distance learning is that it is sufficient to take a course syllabus and course material from on-campus teaching and deliver it to students via CDS or WWW (Bourdeau and Bates, 1996). However, one needs to understand that the multimedia courseware development requires a methodology suitable for the specific subject to be taught, the audience to be addresses and its suitability for a specific learning environment (Retalis, 1997). It requires the developers to undertake a number of related activities for the development of effective, successful and quality courseware. It is believed that, like traditional software development, one should use a systematic, well-disciplined and practical approach to design and develop a multimedia courseware. A development process model provides the developers with such a systematic and disciplined approach which explicitly shows what is to be done and in which sequence.

Corresponding Author: Dr. T.S.K. Homeed, Department of Computer Science, University of Bahrain, Kingdom of Bahrain

Realizing the need for a systematic approach for courseware development, a number of courseware developed models have been proposed (Bostock, 1996; Grutzner *et al*., 2004; Retalis, 1997; Low *et al*., 2003). All of the proposed models have been adapted from the discipline of software engineering models such as waterfall model, prototyping model and incremental model (Pressman, 2005). Weaknesses and strengths of these models have been discussed (Bostock, 1998; Grutzner *et al*., 2004). However, none of these models has been widely applied and evaluated to give a degree of confidence in using them. Therefore, still most of the courseware development takes place using informal and ad hoc method/models (Cochran and Rodrigo, 2006; Whittington and Sclater, 1998; Margi *et al*.; 2000; Alfonseca and Lara, 2000). The ad hoc and informal methods may prove successful on simple and small size courseware development, the scalability of these methods to large and complex courseware development may result chaos and failure. In 2003, the authors of this study undertook a project to develop a number of multimedia courseware that could be used in distance education as well as an aid to face-to-face education delivery. The courseware was required to be of high quality, easily upgradeable and highly flexible and to be completed within time. The unavailability of widely used and evaluated methods and the weaknesses in the existing models motivated us to propose, use and evaluate a process model suitable for the development of a high quality courseware within time and budget constraint.

In this study, we present a process model to develop multimedia courseware and its application in the development of a number of computer science multimedia courseware. We present a courseware development process model and discuss its various phases and activities in detail. We also present a methodology for courseware quality assurance and its relationship with the development process model. Our experience with developing a number of courseware using the proposed model is also presented.

COURSEWARE DEVELOPMENT PROCESS MODEL

A courseware generally refers to content specific instructional software which functions to generate instruction with the support of instructional delivery system. A courseware product involves five elements: the contents and the learning/pedagogical methods as its main component, the learning objectives and the medium as its attributes and the architecture which organizes the courseware in a way convenient to use (Zhiting, 1996).

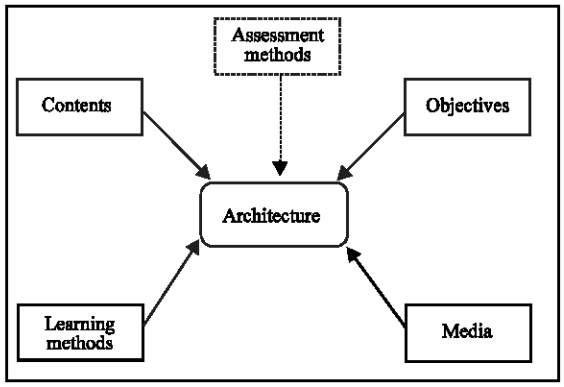

Fig. 1: Structure of courseware

This definition is illustrated graphically in Fig. 1 showing an extra courseware element assessment methods, which we believe should be an important part of a courseware in the context of distance learning. The architecture plays a central role in the courseware and is used to denote the specific way in which other components are organized into a specific courseware product. Also, the contents or the subject matter is the essential component of the courseware; the contents component is linked to some types of learning/pedagogical methods or instructional strategies. The media is the physical basis representing the contents. Since allocation of specific media type (text, audio, video, etc) to pieces of the instructional material should be done, the medium can be regarded as attribute of the courseware. In addition, a well-designed courseware should have a clear specification of learning objectives or goal structure. The assessment methods are used to assess the achievements of a learner so that he can be appropriately directed through the courseware.

Developing multimedia courseware (i.e., contents, methods, media etc.) is a difficult and challenging exercise. It requires the developers to undertake a number of related activities for the development of effective, successful and quality courseware. It is believed that, like traditional software development, one should use a systematic, well-disciplined and practical approach to design and develop a multimedia courseware. A development process model provides the developers with such a systematic and disciplined approach which explicitly shows what is to be done and in which sequence. There are a number of courseware process models that have been proposed in the literature (Lee and Sullivan, 1996; Bradler, 1999; Allesi and Trollop, 1991; Bostock, 1996; Koper, 1995). However, there is no single recipe for courseware development which is either universally accepted or could be used in all circumstance (Bostock, 1996).

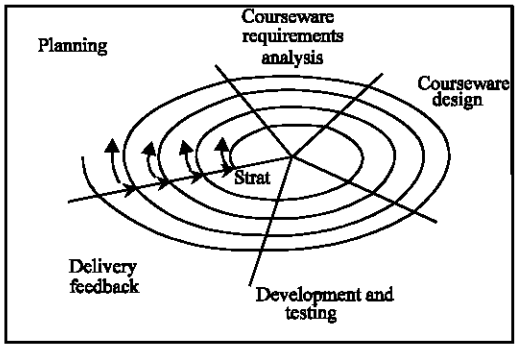

Fig. 2: Courseware development process

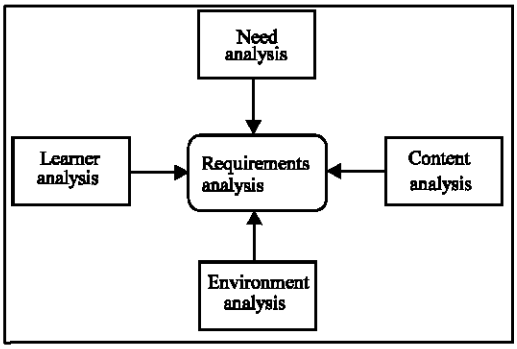

Fig. 3: Types of analysis

We present a variation of spiral model (Boehm, 1988) for courseware development. The main reason for using a spiral model is the fact that the courseware development is a highly iterative process which can well be illustrated by spiral model. It allows courseware to evolve and absorb the changes in the contents and presentation of material with time. The major phases of the propped model are shown in Fig. 2. We would like to emphasize that the spiral model given in Fig. 2 provides a general framework for courseware development. Each of the phases should be carried out using other suitable processes as is discussed in the following sections.

Planning: Like any software engineering project, a successful courseware development project requires careful planning. The development team must understand and specify the scope of the work to be done, the required resources, the risks to be incurred, the tasks to be accomplished, the effort (cost) to be expected and the schedule to be followed. Unfortunately, no widely accepted methods and tools exist for managing and planning courseware development. However, many effective project planning methods and techniques have been proposed in the software engineering literature of which can be used for courseware development planning. For example, bar charts which graphically illustrate the work plan and project *milestones* can be used. Also, cost estimation techniques (Papasyrou *et al*., 1997; Marshall *et al*., 1994) and the quality management techniques (Grutzner *et al*., 2004) have specially been proposed for courseware engineering which can be used during this phase. It is of utmost importance that the management plan should be written down which should included details of the objectives of the project, work schedule, deliverables, cost estimation and so on.

Courseware requirements analysis: After doing a careful planning, development of multimedia courseware begins with the analysis phase. Analysis is the process of defining what is to be learned by the students in a course in a given environment. There are four different types of analysis that should be carried out: need analysis, learner analysis, content analysis and environment analysis (Fig. 3).

The first step in the systematic design of courseware often begins with analyzing the underlying needs and problems in instructional delivery. Need analysis enables identification of problems related to teaching and learning that are specific to content of a course (Bourdeau and Bates, 1996; Zhiting, 1996; Allesi and Trollop, 1991; Grutzner *et al*., 2004). The data is generally gathered from the real environment and the results are used to determine if the multimedia courseware can be used to solve problems in the conventional lectures. The need analysis helps to identify the skills and knowledge lacking in the students that can be addressed in the courseware. This is easily done by feedback from students and academicians.

The purpose of learner analysis is to identify general characteristics of the audience or students, their readiness for the course and their preferences (learning styles) and limitations. This done using diagnostics test (or pretest) to determine students' level of subject expertise (Carver *et al*., 1999; Low *et al*., 2003). Audience skills that are needed in using multimedia courseware, such as technical expertise in using Web as a tool, computer skills, study skills, communication skills and language skills, are all vital in ensuring the effectiveness of courseware. Learner analysis provides a better understanding of students to enable better planning for learner-centered program and to better guide students to process online information.

The purpose of content analysis is to identify topics and subtopics for a course. It helps in identifying the prerequisite skills required for the students to learn the

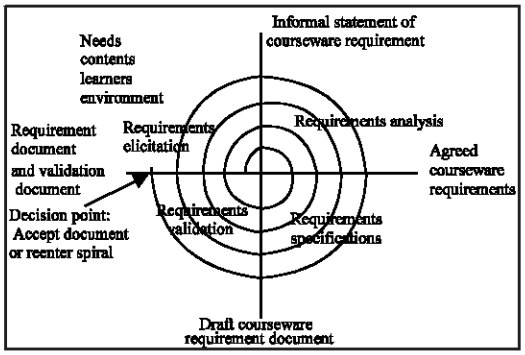

Fig. 4: Courseware requirements analysis process

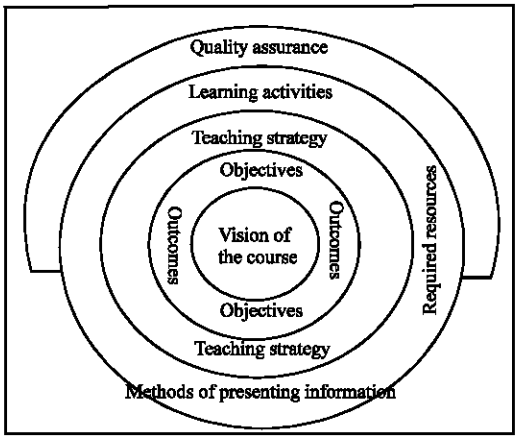

Fig. 5: Content design framework activities

course effectively. It allows for the identification of relationships between topics and for the selection of the right contents for the tasks and skills necessary to learn them (Carver *et al*., 1999; Lee and Sullivan, 1996; Low *et al*., 2003). The relevant information sources are then gathered from textbooks and references to achieve the intended learning outcomes. A closer investigation on the key learning concepts and contents to be taught has to be identified. Note that the contents for a particular course might have been already defined in the curriculum. A careful analysis of the contents has to be carried out to discover the contents that can be learned by discovery or peer interaction and the part of the contents that cannot be learned using multimedia courseware or other resources other than by having an interaction between students and teachers. Therefore, a decision is made on which part of the course contents should be constructed. Task analysis and contents analysis result in a listing of learning objectives, learning tasks, subtask and contents scope that enables the arrangement of the courseware into relevant modules, units and topics.

The purpose of environment analysis is to gather the information about the learning environment and its affect on the learner. It also involves technological analysis to discover the strengths and weakness of available technologies and infrastructures to be used in delivering the courseware. The courseware analysis should be properly documented in courseware requirements specification document.

The courseware requirements analysis should be conducted through a well-defined process. Figure 4 shows, conceptually, how these activities comprise an iterative requirements engineering process. The different activities in courseware requirements analysis (Fig. 4) are repeated until an acceptable courseware requirements specification document is produced or until external factors such as schedule pressure or lack of resources cause the process to terminate. Requirements elicitation is the first stage in building an understanding of the need contents, environment and learner characteristics. It is fundamentally a human activity and is where the audience (learners (students), instructors, etc.) are identified and relationships established between the courseware development team and the audience. The outcome of this activity is informal courseware requirements statement. During the requirements analysis phase, we detect and resolve conflicts between requirements, discover the bounds of the courseware and how it must interact with its environment and elaborate courseware requirements. This culminates into an agreed courseware requirements. From courseware requirements, requirements specifications are derived and they are validated and verified.

Courseware design: Course design is the process of specifying how the contents are to be learned and presented (Allesi and Trollop, 1991). The system is closely examined to design a learning program to be delivered that will ensure mastery of the competencies in the area of knowledge, skills and attitudes (Shih and Davis, 1997). In a multimedia courseware development, various types of design are required. Content design is one of the most crucial design activities. The major content design activities are shown in Fig. 5. The purpose of content design is to identify important information about the course to be communicated to students. The contents have to be properly defined and presented in an effective manner (contents architecture). The content delivery strategies (Gagne, 1985) and media selection should also be outlined during this phase. This serves to direct the learners' attention to the expected learning outcomes and desirable performance. In order to achieve the expectation from the students, the objectives are classified according to levels of learning and to set the

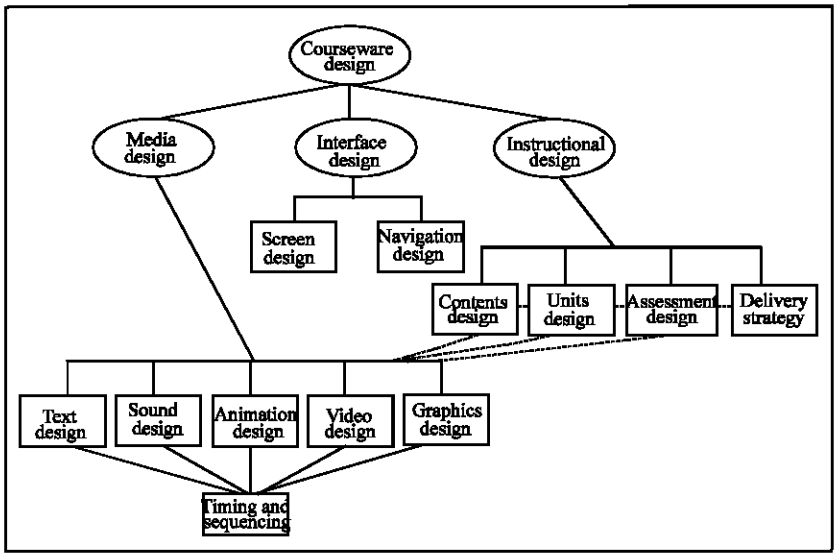

Fig. 6: Design activities and interdependencies

criteria for measuring the learning outcomes. During the content design, the developers must give the quality assurance a central role. The quality assurance activity helps achieving high quality contents. User interface design, layout, navigation etc are also important design activities. Table 1 summarizes various design activities and sub-activities to be carried out in each one of them and Fig. 6 shows dependencies of some of the activities (shown as dotted lines). Timing and sequencing of contents and related multimedia objects must be considered, so that the sound and other media match each other.

Development and testing: The analysis specification and design specification guide us to the next development phase-courseware development and testing. Development is the process of authoring and producing the learning material. The testing is preformed to make sure that the courseware has been developed according to the specification and meets the requirements. There are various activities that are carried out during this phase. Authoring of the course contents is, perhaps, the most important activity during which the contents of the course are written in a way that can achieve the objectives of the courses and intended learning outcomes. During the development phase, graphics, sound, animation and video are developed. Once various courseware components are developed, they are integrated into a whole. Testing is a major activity which can begin as early as the production of individual components is complete (unit testing). Once all the components are integrated, the whole courseware needs to be tested using various integration testing methods (Pressman, 2005). Major activities and sub-activities of this phase are shown in Table 2.

Delivery and feedback: Once the courseware is developed and tested, it can be delivered to the users (tutors, students). A continuous feedback is required from the users to maintain the system. The feedback helps in correction of errors, perfection of courseware and adaptation for new contents and media elements. A very elaborate process should be in place to receive the feedback from the users. Based on the result of feedback, we can continue with the next iteration of spiral.

Courseware quality assurance: A good courseware development process/approach must support continuous quality assurance of all development artifacts and the courseware to be produces itself. Most of the existing courseware quality assurance approaches propose only evaluations of the fully implemented courseware. The major drawback of this late quality assurance is that solving problems introduced during the requirements analysis, specification and design phase are much more expensive than solving them in an early phase when they are introduced (Grutzner *et al*., 2004). Even the prototype quality assurance approach (Driscoll, 1998) assures the quality of only a few artifacts at early development phases. Therefore, it is important to have a life-cycle quality assurance approach that covers all development artifacts and the final courseware as is proposed in software engineering. The quality of artifacts that are produced during each phase is assured using four different methods. These methods include perspective-

Table 1: Design activities and sub-activities

Activities	Sub-activities
Instructional design	Allocation of contents to courseware parts
	Allocation of learning activities to courseware parts
	Courseware unit design, learning step specification
	Media selection for each component of courseware
	Design of assessment method
	Content delivery strategy design
Media Design	Design of text
	Design of graphics
	Design of sound
	Design of animation
	Design of video
User-interface design	Navigation design
	Screen design

Table 2: Development and testing activities and sub-activities

Activities	Sub-activities
Contents development	Authoring of course contents
	Definition of links glossary entries etc
	Development of assignments
	Development of quizzes, exams etc.
Multimedia development	Authoring of script and sound recording for voice over
	Preparation of graphics
	Preparation of animation
	Preparation of video
User-interface development	User-interface implementation
	Screen and interaction template implementation
	Insertion of multimedia components (sound, graphics etc) as appropriate places
Courseware integration	Integration of units to make a whole
Testing	Unit testing
	Integration testing
	User-interface testing
	Pilot testing with real tutors, students

based inspections, prototyping, tests and both formative and summative methods. The responsibility of quality assurance rests with different people who play various roles during the life-cycle, such as Project Manager (PM), quality manager, instructional courseware designer (ID), subject matter expert (SE), Courseware Author (CA), programmer, multimedia developer, graphics designer and artist, video recording expert, potential learners, tutors and so on. This is the approach that was used in our courseware development process.

The output artifacts of analysis and design phase are verified against the output of the previous phases by means of perspective based inspections. Each participating role checks whether the artifact (e.g., requirements specification, courseware design) meets the definitions and specifications of the previous phase. The problem statement and planning documents produced during the planning phase is inspected by project manager, learner, instructional courseware designer, subject matter expert and courseware programmer to ensure that the objectives and scope of the work, the required resources and schedule are realistic and can be achieved. The requirements specification produced during the analysis phases are inspected by all the roles mentioned before to ensure that the requirements specifications meet the goals and objectives of the courseware as outlined in the planning document. Note that the requirements verification and validation is an integral part of the requirements analysis process model shown in Fig. 4. The courseware design specifications produced during the design phase are validated against the requirements specification. This activity is also performed by all the people involved in the development of courseware. The errors and problems that are discovered during the inspections are collected and discussed in an inspection meeting. Errors are corrected and, if necessary, another inspection cycle might be started. Our experience with the development of courseware shows that inspection based method coupled with prototyping and the participation of various roles at every stage of the project facilitates an agreement about the current development of the product among the project members and provides a very valuable input to improve the quality of the end-product.

The design artifacts which are produced as a result of different activities as given in Table 1 can be verified against the design specification using inspections and prototyping of certain aspects of the courseware. This helps in achieving a high-quality product. Once the courseware is developed, it can be verified against the design specification and by testing the individual components and courseware as a whole. The test cases for testing are generated during requirements specification, courseware unit design and during the specification of functional requirements. A test case contains instructions of activities that have to be performed with the courseware as well as expected results or system behavior. All the tests are organized and supervised by the quality manager along with development team; they specify the test cases to be applied and compare the test results with the expected results from the test case specification. If there are any deviations, they identify the problems behind the deviations and make arrangements in order to solve them. A test cycle ends when all test cases run properly without any deviation from the expected results. Summative evaluation is performed to deduce the effectiveness of the courseware and the mode of instructional delivery. A commonly used approach for summative evaluation is the pre-test and post-test (Retalis, 1997). Two types of questionnaires should be given to the learners. The first one concerns the pre-test which aims at identifying the expectations of the learners as well as organizational matters of the instructional delivery. These questionnaires should be filled in during the first days of the courseware delivery and not latter than the first week. The post-test

questionnaire helps to deduce overall judgments and criticism on the courseware, learning environment and instructional delivery. The questionnaires should be collected for analysis, interpretation and reports of the results.

Application and evaluation of the process model: The courseware development process presented in to develop several coursewares for different undergraduate and postgraduate courses. In each project, initial requirements analysis was carried out by a team of faculty members of the department of computer science. A survey of various other universities already offering that course was also carried out to determine the learners' characteristics, needs and learning outcomes. Keeping in view the results of our surveys and discussions with students and teachers, existing course description was revised so that the intended learning outcomes of the course could be achieved. Topics and subtopics and their relationship to learning outcomes were then established. The course specification prepared was reviewed by a committee of experts in each course to get their feedback on course topics, subtopics, objectives and so on. The course specification was then revised again keeping in view the feedback provided the reviewers. This helped us in precisely outlining the courseware description so that the intended learning outcomes could be achieved.

During the instructional design phase, we carefully divided the contents into a number of units and learning objectives and outcomes were assigned to each unit as determined during the analysis phase. Using the Gagne (1985) description of nine external instructional events aimed at facilitating the acquisition of different kinds of intellectual skills in a systematic way, we derived a content delivery strategy for each unit of the courseware. The learning strategy aimed at the ways of gaining the learner attention, informing the learner of the objectives and outcomes of each unit and its relationship to overall course objectives, presenting the important points and concepts in a distinctive manner, proving learning guidelines, presenting the assessment methodology, providing the feedback aimed at reinforcement of learning and so on. We also planned and designed the relevant graphics, sound/voice requirements and animation suitable for various courseware components. An easy to use and consistent user-interface was also designed. The whole designed was reviewed by a team of developers, content writer and project director and necessary improvements were made before starting the implementation following the activities and process model describe earlier.

Once the revision of design was found to be adequate with respect to course objectives, learning outcomes, learning methodology, planning and design suitability and consistency, the development phase began. During the development phase, course contents were prepared. A subject expert or team of experts is given the task of authoring the course contents. Once the course contents were ready, they were reviewed by an independent committee of subject experts for their accuracy, organization, clarity, suitability and its conformance with the course objectives. The authors were asked to make the necessary revisions.

The multimedia aspect of the courseware was limited to text, graphics, animation and sound/voice. Using various technologies (voice recording, digitization and editing, Adobe Photoshop, Corel Draw, scripting using action scripts, HTML, etc), various components of the courseware were produced and each unit was tested individually and then integrated with the other units. Reading material to further the learning process and glossary are also provided on each topic. At the end of each unit, a learner is assessed with a well prepared quiz for self evaluation. The questions in each quiz are prepared in such a way that they test learner's ability to retain important concepts, application of concepts to various situations and skills to be acquired.

The testing of the courseware was performed at the unit level, integration level as well as at course-level by the members of development team, subject experts, a sample of students and external independent testers. The feedback from various testers was discussed among the development team and necessary changes were made before releasing the courseware on the internet and CDS.

We also evaluated how successful the proposed model has been in achieving our objective of developing of high quality courseware within time and budget constraint. Since there have been no data available on the performance of the existing models to which present results could be compared, the focal question in evaluation of our process model was how do we know and judge that the courseware development process has helped in developing high quality courseware within time and budget. However, it is suggested (Retalis, 1997) that a process method should be evaluated by testing and validating the quality of the courseware and by determining whether the money as well as the time spent for its development were within the predicted limits. Therefore, we evaluated our process model based on this criterion.

To determine whether we were able to complete the courseware development within time and budget, we collected the required data from the project manager and

checked against the management plan. The data shows that five out of total eight projects were completed according to the time set within the minimal deviations which were within the tolerances and the budget of the projects was not exceeded more than a predicted threshold. The data also revealed that course analysis and design were the most time consuming activities followed by the development of multimedia components. The remaining three projects were delayed from three to six months. The investigations revealed that the major reasons for their delays were the lack of expertise in developing the course contents for those particular courses and design and development of multimedia components. The review of content material by external reviewers was another contributing factor who failed to complete the review within the allotted time. However, this situation was improved in the last three projects when the project manager followed up the review process. With out a doubt, our success rate in developing the courseware within time and budget has been very good by all standards when compared with the courseware and software development success rates reported by others (Pressman, 2005; Retalis, 1997). We believe that our success rate will further improve as we get more experience in using the proposed model and available technologies as is the case with software engineering.

To determine the quality of the courseware, we used the guidelines provided by Retalis. (1997). We collected data using the formative evaluation questionnaires and the pre-test questionnaires from subject experts, tutors and learners for three courses. The data show that the quality of the courseware checked against the standards is quite high. The initial summative data collected from the tutors and students show that the courseware has been found very useful by the students and tutors. 67% respondents was fully satisfied with the quality of the courseware and 25% were partially satisfied and pointed out some areas of improvements both in the contents and delivery method. Remaining 8% suggested major improvements and they were not satisfied with major portion of the courseware. Feedback from the tutors shows that with the use of the courseware, students learning time was reduced by 10-15%. These results are comparable or better than those reported in the literature (Papaspyrou *et al*., 1999; Whittington and Scalter, 1997; Kunst *et al*., 1997; Johnson *et al*., 2004; Margi *et al*., 2000; Cochran and Rodrigo, 2006). Furthermore, the application of the quality assurance methodology showed that performing inspections of the early products/artifacts revealed many errors and problems, which were removed with minimum of effort. This was also evident from the fact that only 5-10% changes/corrections were suggested by the independent reviewers when they evaluated the completed courseware. However, more detailed evaluation of various aspects of multimedia courseware is being conducted to further improve both the development process and the product. However, complete data will not be available until the mid of next year. As a future work, we intend to improve the courseware once we get all the feedback and complete the performance evaluation study.

Based on our evaluation results and experience with the proposed model, we conclude that the proposed courseware development process provides developers with a systematic and formal methodology that could be used in developing high quality, easily maintainable and flexible courseware. Since the proposed process provides a general framework for courseware development, it can be tuned up according to the specific preferences and needs of the developer. In a sense, it can be seen as a generic courseware development methodology. In addition, the propose model is scalable as has been proved in our project of developing courseware of various sizes and complexities.

CONCLUSIONS

In the context of distance learning, multimedia and WWW provide new opportunities and mechanism for developing course material and its delivery to students. The process of developing multimedia courseware is an area that requires careful consideration as technologies and tools for both the authoring and the delivery of courses are evolving rapidly. In this study, we presented a multimedia courseware development process model and various activities that are undertaken to successfully develop and deploy a multimedia courseware in distance learning paradigm. The courseware quality assurance is a very important activity that needs to be carried out throughout the life-cycle using various methods such as inspections, reviews, prototyping, pre-test and post-tests. The experience so far has been very rewarding and the feedback received so far from students, tutors and other involved in the courseware design has been encouraging. The initial feedback from the students and teachers shows that the courseware, to a large degree, achieved its intended goals of proving consistent high quality education. However, further research is required to explore how the learners and instructors can work together to develop technology goals and planning guides through on-line and offline activities. Some areas which need more research are standard-based activities, on-line assessment, on-line collaborative projects and assignment, virtual field trips, ways to make courseware more interactive and responsive to the needs of individual

leaner, providing class-room feeling while one is learning using courseware etc.

REFERENCES

Alfonseca, M. and J.D. Lara, 2001. Automatic generation of simulation-based web courses and model documentation. Transaction, 17: 106-111.

Allesi, S. and S.R. Trollop, 1991. Computer-Based Instruction. NJ: Prentice Hall.

Boehm, B., 1988. The spiral model for software development and enhancement. Computer, 21: 61-72.

Bostock, S., 1996. Courseware engineering: an overview of courseware development process. Available at www.keele.ac.uk/depts/cs/stephen_bostock/docs/atceng.html.

Bourdeau, J. and A. Bates, 1996. Instructional design for distance learning. in Proceedings of 4th International Conference in Intelligent Tutoring Systems, pp: 27-34.

Bradler, J., 1999. Developing on-line learning materials for higher education: An overview of current issues. Educational Technol. Soc., 2: 112-121.

Carver, C.A. and M.A. Biehler, 1994. Incorporating multimedia and hypertext documents in an undergraduate curriculum. in Proceedings of IEEE/ASEE Frontiers in Education, pp: 87-92.

Carver, C.A. R.A. Howard and W.D. Lane, 1999. Enhancing student learning through hypermedia courseware and incorporation of student learning styles. IEEE Transactions on Education, 42: 33-38.

Cochran, S.K.M. and R.L. Rodrigo, 2006. Determining effective distance learning designs through usability testing. Computer and Composition, 23: 91-107.

Driscoll, M., 1998. Web-Based training: Using technology to design adult learning experiences, Jossey-Bass/Pfeiffer, San Francesco.

Gagne, R.M., 1985. The Conditions of learning, 4th ed NY: Holt, Rinehart and Winston.

Grutzner, I., S. Weibelzahl and P. Waterson, 2004. Improving courseware quality through life-cycle encompassing quality assurance. in ACM Symposium on Applied Computing, pp: 946-951.

Johnson, T.R. and J. Zhang *et al.*, 2004. Assessing informatics students' satisfaction with a web-based courseware system. Intl. J. Med. Inform., 73: 181-187.

Koper, R., 1995. PROFIL: A method for the development of multimedia courseware. BJET., pp: 26: 94-108.

Kunst, S. and C. Mast *et al.*, 1997. Theaching multidisciplinary teamwork. Proceedings SEFI Annual Conference on Humanities and Arts in a Balanced Engineering Education, pp: 108-111.

Lee, P.M. and W.G. Sullivan, 1996. Developing and implementing interactive multimedia in education. IEEE Transactions on Education, 39: 430-435.

Low, A.L., K.L. Low and V.C. Koo, 2003. Multimedia learning systems: a future interactive development tool. Internet and Higher Education, 6: 25-40.

Margi, C.B., Vilcachague *et al.*, 2000. An online web course environment and its application. 30th ASEE/IEEE Frontiers in Education Conference, pp: 1-6.

Marshall, I.M. and W.B. Samson *et al.*, 1994. Predicting the development efforts of multimedia courseware. Information and Software Technol., 36: 251-258.

Papaspyrou, N. and S. Retalis *et al.*, 1997. A model for multimedia courseware cost estimation. in Proceedings of the 3rd Panhellenic Conference with International Participation on the Didactics of Mathematics and Informatics in Education, pp: 417-436.

Papaspyrou, N. and S. Retalis *et al.*, 1999. Web-based teaching in software engineering. Advances in Software Engineering, 30: 901-906.

Pressman, R.S., 2005. Software engineering, 6th Edn., NY: McGraw Hill.

Retalis, S., 1997. A courseware development methodology for open and distance learning. 4th Doctoral Consortium on Advanced Information System Engineering, http://www.uni-koblenz.de/fb4/publikationen/ gelbereihe/ RR-14-97/retalis.ps.gz.

Shih, T.K. and Davis, R.E., 1997. IMMPS: A multimedia presentation design system. IEEE Multimedia, pp: 67-78.

Whittington, C.D. and N. Scalter, 1997. Building and testing a virtual University. Computer Education, 30: 41-47.

Zhiting, Z., 1996. Cross-cultural portability of educational software: A communication-oriented approach. Ph. D Thesis, University of Twente.