

A Distributed Online Curriculum and Courseware Development Model

Pinar Onay Durdu

Kocaeli University, Department of Computer Engineering, 41040, Kocaeli, Turkey //
pinar.onaydurdu@kocaeli.edu.tr

Nese Yalabik

Middle East Technical University, Department of Computer Engineering, 06531, Ankara, Turkey //
yalabik@ceng.metu.edu.tr

Kursat Cagiltay

Middle East Technical University, Department of Computer Education and Instructional Technology, 06531, Ankara, Turkey // kursat@metu.edu.tr

ABSTRACT

A distributed online curriculum and courseware development model (DONC²) is developed and tested in this study. Courseware development teams which may work in different institutions who need to develop high quality, reduced cost, on time products will be the users of DONC². The related features from the disciplines of instructional design and software engineering were combined. The research is conducted as a collective case study, including four cases with distinctive characteristics to reveal the several practices in online curriculum and courseware development work. The DONC² development model was proposed using the results gathered from the investigated cases and a literature survey. The model uses the iterative incremental and agile software development approaches in order to overcome the disadvantages of other linear development approaches. This enables building releasable yet good quality products in short time periods. Furthermore, continuous communication, evaluation and feedback as well as good project management and readiness to adapt to changes are integrated as the essential characteristics. DONC² is different than previous linear and non-adaptive models in all of these aspects. The model was tested with one of the cases while being improved with success. It was applied as a development model for scheduling of the courseware development project in the last case and considered as helpful for the success of the project by the project team members.

Keywords

Virtual learning environments, Online courseware and curriculum development, E-learning, Adaptive software development

Introduction

The number of people who are seeking a university degree, skill enhancements or lifelong learning has increased tremendously. This has forced universities and companies to find new ways to provide education to the mass learners and recent developments in information technology and Internet have enabled that by delivering web-based courses via “virtual learning environments” (VLE) (Xu, Wang & Wang, 2005, p.525). Many institutions have started projects to employ e-learning, which has the goal of “learning anytime and from any place” (Barjis, 2003, p.4).

VLEs are the online learning systems which provide a complete learning environment including various features such as course materials, evaluation instruments or communication and collaboration tools (Ryan, Scott, Freeman & Patel, 2000). They can be considered as the new form of providing education. Their features should be very different from the traditional classroom settings since there is no face to face interaction between the instructor and the students. On the other hand, VLEs may provide many additional opportunities for achieving enhanced and enriched learning outcomes through the use of the web for effective instruction and can be a promising alternative to traditional settings (Zhang, Zhao, Zhou & Nunamaker, 2004). Web-based teaching can facilitate learner interactivity and also can provide a great amount of resources. In order to enhance learner interaction, courses can use innovative and dynamic learning materials. Unfortunately, these advantages are still not being used effectively since usually traditional instructional design methods are transferred to the web as if they were in the traditional settings. In addition, there is still no commonly accepted framework to guide developers in their design of curricula (Oliver & Mcloughlin, 1999). On the other hand, collaboration of a number of institutions might be important for the use of best resources in online material development and teaching.

For the success of VLEs, the effectiveness of courseware provided is essential. In order to provide effective courseware, careful consideration is needed during the development stages with the help of several disciplines such as instructional design, software engineering and human computer interaction (HCI).

ADDIE (analyze, design, develop, implement and evaluate) is a prescriptive sequential instructional design model, which describes the essential components of any instructional development process. Many other models that use this classical sequential approach have been developed. The ‘Instructional Design Plan’ is a relatively new model for individual course development. It was developed by Kemp, Morrison and Ross (2004) and is different from ADDIE since it does not apply any specific sequence and considers the design and development process as a continuous cycle as can be seen in Figure 1. This model also covers issues like project management or implementation in addition to the instructional design steps.

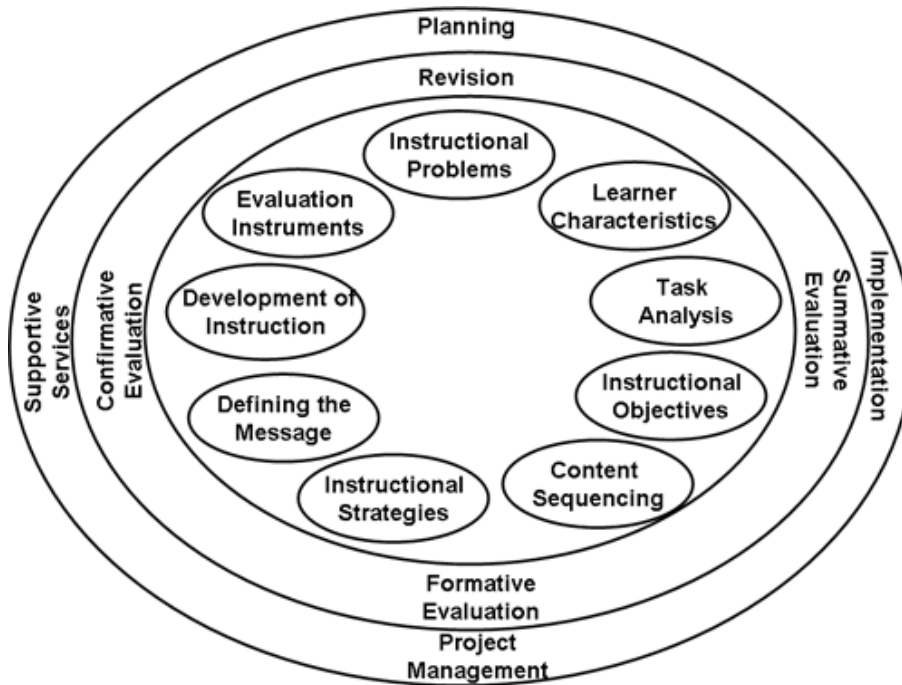


Figure 1. Instructional Design Plan (Kemp, Morrison & Ross, 2004, p.1)

Some of the instructional design models are adapted to online course development (Tripp & Bichelmeyer, 1990, Willis, 1995, 2000). ‘Rapid prototyping’, proposed by Tripp and Bichelmeyer (1990) is an example. It does not use a sequential approach either. It is the best known model that makes use of the software engineering methodologies and the steps are realized almost in parallel with iterative development as seen in Figure 2.

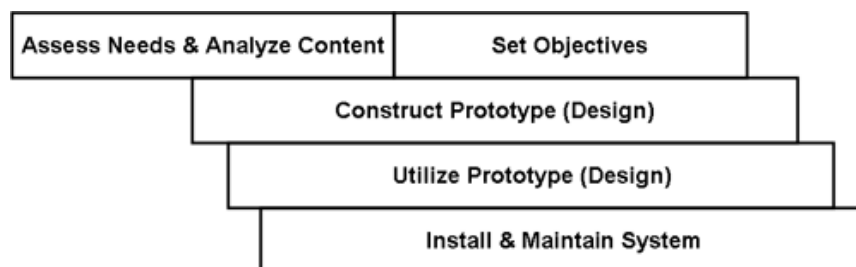


Figure 2. Rapid prototyping approach adapted to Instructional Design (Tripp& Bichelmeyer, 1990)

Developing educational software has many common aspects with software development. Especially the design and production stages are similar since the product as well as the production medium and tools are the same for both.

Recently, software practices which are commercially proven to be successful in the software engineering field introduced iterative and agile methods. These focus on iterative development, managing requirements, using component based architectures, visually modeling software, verifying software quality and configuration management (Kruchten, 1998)

Agile methods call for lighter weight and faster development methods. Although these methods are sometimes considered as ad hoc or unstructured, they involve balanced structure and flexibility in their core, which is claimed to increase creativity and innovation (Highsmith, 2002). The important features that differentiate agile development from former models can be summarized as (Abrahamsson, Salo, Ronkainen & Warsta, 2002):

- The emphasis is on building releasable software in short time periods.
- The time period used for iterations is measured in weeks rather than months.
- They focus only on the functions required at first hand, delivering them fast, collecting feedback and reacting to received information.
- Close team relationships, close working environment arrangements and procedures that improve team spirit are at the core of the models.
- Continuously producing tested working software, collaboration with the customer and responding change.

Adaptive Software Development (ASD) (Highsmith, 2000) is an example framework among the agile approaches which emphasizes project management and collaboration practices (Highsmith, 2002). It is going to be explained in Section 2 in more detail.

Incorporating HCI, more specifically the usability issues to the development approach is essential for the overall quality of learning since the user will judge the system on the basis of interface which is the first contact point (Faulkner, 1998). Usability ensures the ease of learnability of the learning environment as well as the learning content, which increases the effectiveness. Therefore, in VLEs, since the core aim is to learn the contents rather than learn to use the system, the more usable the system is the more effective the learning takes place. Standard ISO 9241 defines usability as “allowing the user to execute his task effectively, efficiently and with satisfaction in the specified context of use” (Abran, Khelifi, Suryan, Seffah, 2003, p.331). In addition to that, usability approaches are also considered since they focus on formative and summative evaluation methods needed to be implemented throughout the whole development process in order to ensure the effectiveness of an instructional product (Crowther, Keller, Waddoups, 2004).

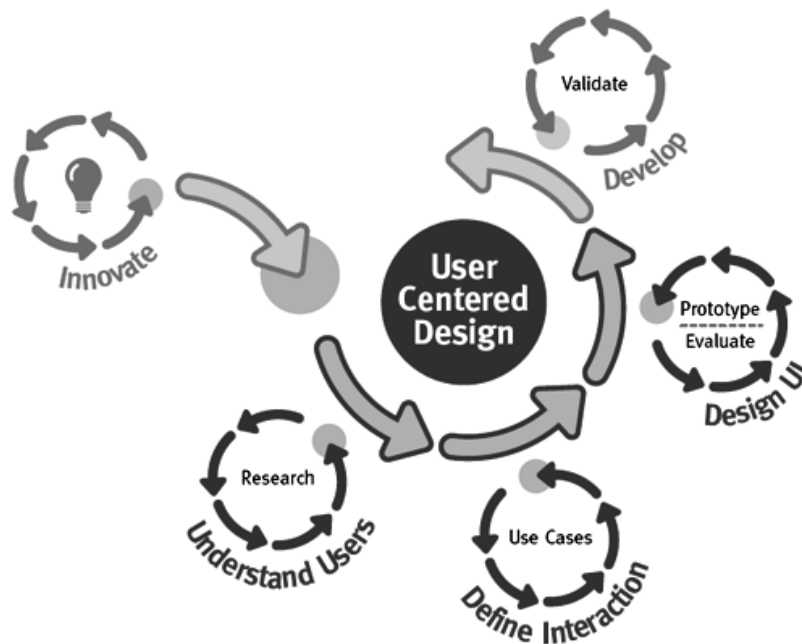


Figure 3. Three Iterative UCD Phases (Detweiler, 2007, p. 41)

Usability issues or concerns are not systematically covered in many software development approaches as there are no known methods to integrate these concepts to the development life cycle. HCI issues are generally considered only at the screen-interface or at the final design stages (Zhang, Carey, Te'eni & Tremaine, 2005). On the other hand, many of the software developing organizations are beginning to pay more attention to the usability of their products. In addition they also realize the importance of implementing these techniques early in the development processes (Ferre, 2003; Anderson, Fleek, Garrity & Drake, 2001). In a more recent study, Detweiler (2007) proposed three iterative phases which are repeated in all development phases of an agile software development project. This approach enables users to test the system from the beginning to the end of the development effort and to design, prototype and develop user interfaces iteratively, as seen in Figure 3.

The importance of usability is also realized in instructional development field. In addition to traditional evaluation done generally at the end of development cycle, the need to incorporate usability evaluation from the beginning to end is has been acknowledged by many instructional designers. Therefore, systematic formative and summative evaluation methods are needed to be implemented throughout the whole development process in order to ensure the effectiveness of an instructional product (Crowther, Keller & Waddoups, 2004).

Another important issue in the development of educational software is the use of related industry standards. Technology enhanced learning environments are requested to be compliant with current learning technology standards such as IEEE LTSC, AICC, and IMS (Anido, et al., 2002).

The literature review reveals that an integrated development model for distributed curriculum and courseware development for online learning environments has not been studied. In the present study, a collaborative online curriculum development methodology based on the disciplines discussed above (instructional design, software engineering, HCI) is proposed.

The ASD approach mentioned above is adapted as the software development methodology. ASD is described in detail in the next section. The design method used in this study, the outcomes of the investigated cases and the developed methodology and conclusions are presented in the subsequent parts.

Adaptive Software Development (ASD) and Its Adaptation

ASD is an iterative development model that stemmed from rapid application development. However, it is primarily different from them in that its acceptance of emergent order. It provides a dynamic speculate-collaborate-learn life cycle which is different from static plan-design-build life cycle models. It uses iterative cycles like spiral or evolutionary development but it also has another dimension which are represented by secondary arrows leading away from the iterative cycle, as can be seen in Figure 4. These indicate the emergent ideas that can be revealed throughout the development process. In complex environments it is usually not possible to determine all specifications at the beginning. Therefore speculation is offered as a replacement. Collaboration is required to balance unpredictable and predictable specifications. Learning occurs as a result of this collaboration. Stakeholders make small mistakes based on their false assumptions and they learn from their mistakes and gather better experience and mastery. As a result, this dynamic cycle enables continuous learning and adaptation to the emergent situations (Highsmith, 2002).

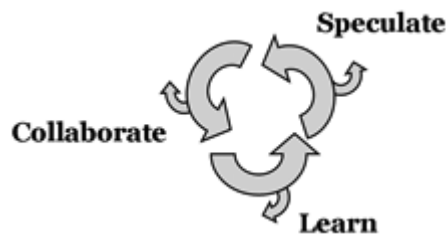


Figure 4. The Adaptive Development Life Cycle (Highsmith, 2002, p.41)

The ASD emphasizes adaptation rather than optimization. It assumes that change and flexibility is necessary, so change management is at the core of the development methodology. It is also a component-based rather than a task-based approach and components are conducted in adaptive cycles throughout the project. The features of adaptive cycles involve the characteristics of ‘mission-focused, feature-based, iterative, time-boxed, risk-driven and change-tolerant’ (Highsmith, 2002, p.83). These short time-boxed iterative development cycles are primarily different than the sequential life cycles in that all life cycle activities are performed throughout the whole life cycle but with different workloads not only at corresponding phases as can be seen in Figure 5. Time-box representation looks similar to Gantt charts but it is actually different. It represents the amount effort for a particular task relative to other tasks. Wider the black area, more time/effort needs to be spent for the particular task. Since the horizontal axis represents time, the overlapped effort among tasks can also be observed. It also shows the amount of concurrency among tasks.

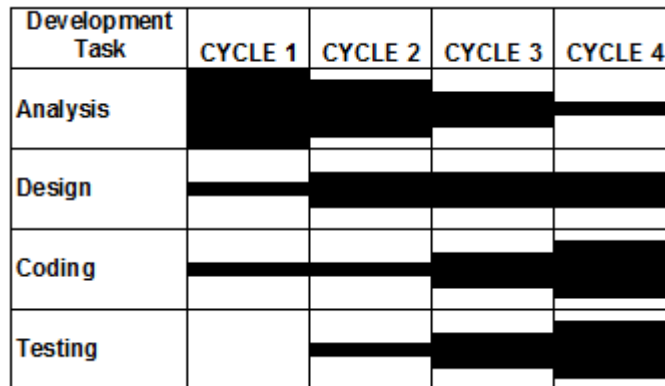


Figure 5. Span of development tasks into development cycles

Change is considered as a likely event for online curriculum and courseware development effort so the model emphasizes continuous evaluation and revision. Importance attributed to continuous collaboration and communication is the other essential feature of the model. It does not provide a rigid structure for development efforts, instead, it focuses on flexibility and guides the developers how to conduct all these components. Since the ASD provides a framework which corresponds to these issues and emphasizes more on project management and collaboration practices, it is used as primarily adopted software development approach for the DONC2 model.

Design of the Study

An iterative incremental approach was also in the design of the study. In the first iteration, two online courseware development projects were investigated. At the end of this iteration a draft version of the development model was defined. In the next two iterations, validation activities for the proposed model took place and the final form of the model was established after the necessary revisions.

Online courseware development projects were examined as cases to see the problematic points as well as best practices that can be applied to the proposed development model. Some criteria which can be seen in Table 1 were determined for the selection of the cases but not all of them could be satisfied by all the cases.

Table 1. Case selection criteria and their occurrences in the selected cases

Criteria	Case 1	Case 2	Case 3	Case 4
Curriculum development focus	-	√	-	√
Online course material development	√	-	√	√
Geographically distributed development environment	√	√	-	-
Teamwork	√	√	√	√

Data collection methods

Interviews were conducted with experts who participated in the projects examined with different roles, to learn about their experiences. In addition to the interviews, the courses developed in the project by different people were also examined to see whether the applied project practices lead to successful results.

Interviews

Three different interview question sets based on the nature of the investigated cases were used throughout the study. The first interview question set consisted of open-ended questions for the semi-structured interview sessions. In those interview sessions, the researcher tried to gather as much information as possible about the reactions of the team members for the courseware development processes. The interview questions were grouped into two categories. The first category involved the general questions, which tried to gather general information about the project as well as the interviewees. The second category involved main questions related to the development process, starting from the strategic decision making to the lowest level activities. These main questions can be classified into three layers.

- *Management layer*: This layer deals mainly with the Project management level strategic decision making activities as well as issues which are directly related with the project manager or project management team.
- *Integration layer*: This layer mainly acts as a middle layer among the management layer and the micro layer. It deals mainly with the curriculum development activities.
- *Micro level*: This layer is mainly deals with the courseware development level activities. It includes all issues to be considered for the development of an individual courseware.

The second interview question set was an evaluation matrix (EM) which was developed based on the proposed development model. This EM was applied to a developed courseware project as an evaluation framework in a structured interview session. There were four matrices covering the three layers of development processes as given above in addition to interlayer processes of communication and evaluation and revision. Matrices include all the elements that should be integrated to the development effort and check whether they were realized in the investigated project as well as at what level they were carried out.

The third interview set was again formed of open-ended questions for the semi-structured interview session. Here, the researcher tried to gather information about the developer's views on the processes applied to the project.

Implementation Process

The implementations procedure and the investigated cases will be explained by giving background information about the subjects as well as the scope of the projects. The cases were studied in three iterations:

Iteration 1

In the first iteration two different instructional system development projects were examined.

Case 1- Avicenna Virtual Campus (AVC) Phase 1 (AVC-1): The major purpose of this project was to create a new community of universities from 14 Mediterranean countries by forming Avicenna Knowledge Centers (AKCs). Each AKC consists of a team of project director, pedagogical expert, technical expert and technicians and these teams has to develop about 20 online course modules in English and in their own language (Avicenna, 2006). The courses developed by the countries are uploaded to a content management platform called "Plei@d" for sharing. This project does not aim to develop any degree program. The phase I of the project was completed at the time of investigation.

Case 2 - SBS- Malaysia Project (SBS): This case was an e-learning project conducted by Siemens Business Services (SBS) for Malaysia Government. The major purpose of this project was to develop computer-aided materials for 7th and 8th grade mathematics and science courses of Malaysian formal elementary school education. It was a collaboration project of geographically distributed teams, which were located in Turkey and Malaysia. In total 330

courses were converted in about sixteen months period successfully. All of these courses have begun to be used by the Malaysian teachers and students at the schools since 2005 (Rtb, 2007).

Iteration 2

In this second iteration, Equipping Primary Care Physicians to Improve Care of Children (EPPICC) case was investigated. The first interview questions were used as in the previous two cases. In addition to this, additional interview sessions were conducted with the EM matrix developed based on the developed methodology.

Case 3 – EPPICC: The EPPICC is one of the online collaborative grant projects developed by the University Of Alabama Division Of CME (Continuing Medical Education) with the collaboration of different departments. It is jointly sponsored by the University of Alabama, Alabama Medicaid, and the National Institutes of Health. The goal of the EPPICC project is to provide online modules to improve the initial screening that pediatricians do with regard to eye care of small children (EPPICC, 2006). The modules are developed by the CME in collaboration with the Principal Investigators (PIs) of the grant. PIs were also responsible for the project management so can be considered as project managers. The CME provides courseware development team and PIs provided the content for the courses. The development of the modules was completed at the time of investigation.

Iteration 3

In this final iteration, the proposed methodology was applied as a development methodology to the Avicenna Virtual Campus Phase 3 project. The project plan was formulated with the appropriate components that are necessary for this phase of the Avicenna project. The project management component from the global layer; training and style guideline components from the macro layer and all micro layer components in addition to the communication and continuous evaluation and revision components were adapted for the project and implementation of the project was conducted based on this. After completion of the project, the third interview set was used.

Case 4 - AVC Phase 3 (AVC-3): In this case, the courseware development effort of the Turkish partner in the last phase of the AVC project was investigated. In this phase six more courses were developed in the scope of the project mentioned in Case 1 by the Turkish AKC. The same requirements were still used as in the previous phases.

Data Analysis

The data analysis was continuous and iterative throughout this work. The researcher first looked for similarities in the data from the participant interviews and the documents of the development effort. Secondly, the researcher looked for data that captured major differences among the efforts. Finally, categories and themes were created to arrive at a list of critical principles.

Outcomes of the Case Studies

The results of the interviews conducted with the team members of different projects and examination of their documents are summarized in the following section. Based on the answers of the interviewees, the major themes necessary for each layer, global, macro and micro layers have been determined.

Interview Results

Project managers, pedagogical experts or software developers who are responsible for strategic decision making level of their projects were interviewed throughout the study. Some principles were extracted and developed from the findings of the cases. These principles are applied in the various stages of the development process, and each has some (more or less) impact on different phases of the development model.

Outcomes of the AVC-1 Case

Providing a good project management team which is supportive, guiding and understanding is essential for the project, as they will enable the coordination among the distributed members is the most prominent principle gathered from this first case. Having people who have required competencies in the team is very essential for good decision-making. On the other hand, technical issues can be distracting during this kind of development projects and can demotivate the teams. Therefore, including a technical support team that will respond the technical needs of development teams in a timely manner is required for the smooth running of the project. Moreover, without adequate participation of any team members, the development approach cannot be succeeded. Training is required for the developers before and during the development process in order those to have the common understanding of the development process as well as what kind of material is going to be produced at the end. In this case, it was revealed that the capabilities of LMS were very limited and produced difficulties. The capabilities of LMS should be determined based on the requirements of the courses to be presented so the selection of LMS is an issue to be concerned during the development effort. Another issues revealed was that there is a need to apply an instructional design and strategies for the development of the courses as the materials are needed to have pedagogically sound principles to aid learning. Furthermore, the developed courses are needed to be evaluated according to their quality. Accreditation mechanisms, quality check, usability reviews are needed to be determined to develop effective learning environments. The finding of the case showed that the course development is not a step-by-step process. In the case, chapters in each course were developed as increments and this prevents the late finding out of errors due to analysis of the courses.

Outcomes of the SBS Case

Several principles were also extracted and developed from the findings of this case. Some of these principles are also similar or complementary with the findings of the first case like providing an effective project management team and a good project manager. Project manager's ability to act as a facilitator, accelerator, motivator or mediator is crucial for the success of the Project. Providing a project plan which is continuously updateable is one of the most emphasized features stated by the interviewees of this case. Since it is not possible to predict everything at the beginning of the project, the need to continuous update of the plan throughout the project is needed to be realized and accepted by all team members. Related to this issue, the requirement of configuration and change management mechanisms was emphasized in this case. Other common principles with the previous case were formation of adequate teams, trainings for team members, quality control process and having an iterative development model. Another distinguished principle mentioned in this case was to have an effective communication infrastructure and mechanisms. Especially in geographically distributed development environments, communication infrastructure which incorporates different communication tools is needed in order to track communication data. Apart from the infrastructure, other mechanisms that define the type and way of communication among all the team members are essential for effectiveness. Rules of engagement should be determined at the beginning of the project.

Outcomes of the EPPICC Case

Based on the results of interviews conducted with the project stakeholders and the evaluation matrices applied in this case some other principles were revealed. Enabling appropriate division of responsibilities in teams is one of them. Moreover, the need for collaborative decision making in teams for easy negotiation among the members as well as enabling awareness of what is going on at any time of the project was emphasized by interviewees. Some distinguishing principles gathered from this case were careful consideration of recruitment and retention, providing a risk plan and providing usability tests. Drawing attention of students to the developed online courses or programs and keeping their attention continuous to them is an important task that should also be considered for the continuity. Planning of publication is crucial. Risk planning makes the project ready to the unexpected events. Since there was no risk planning in this project, schedule slippage occurred due to several reasons. Usability testing will enable the students focus on the content rather that the system. This also helps their retention on the courses without frustration. Trainings and iterative development approach were also emphasized as the outcomes of this case.

Outcomes of the AVC-3 Case

Supportive, directive and concerning project management was also highlighted especially for the motivation of the team members in this case. Providing a project plan which is updateable since the prediction of the later events clearly is not possible at the beginning of the project was also emphasized. In this project this could not be provided so the slippage in the schedule was occurred. People in the development teams may not have the same level of knowledge or experience about online course development as in this case. Therefore, providing training was also stressed in this case. A style guideline determination was emphasized in this case since they provide the unity of the courses that will be served in the same curriculum. This will also enable the assessment of quality of the courses easily. Conducting regular review meetings was considered essential by the interviewees as they believe that they enable the control of the project flow as well as increase the quality of materials

The investigation of all these cases revealed some essential principles that should be included in a development model. These principles especially could be grouped in management or integration layer since they were dealt with management, planning or strategic decision making level activities. Table 2 summarizes all the principles to be included in the model.

Table 2. Summary of the outcomes of all cases

Management Layer	Project management
	Project Managers/Team Leaders
	Continuous update of project plan
	Needs assessment/ requirements analysis
	Partner participation to strategic decision making
	Regular team meetings
	Communication/coordination mechanism/infrastructure
	Conflict resolution mechanism
	Accreditation
	Adequate number of personnel
	Risk planning
	Technical support
Integration Layer	Trainings
	Style guideline
	LMS
	Quality criteria
	Recruitment and retention
Micro Layer	Pedagogical aspects
	Internal reviews as well as external reviews
	Prototyping
	Usability testing
	Revision of the project after courses are given
	Reusability of course materials
Industry standards	

Courseware Examination

In addition to the interviews, the developed course materials were also examined. The courses developed in AVC-1 case were investigated and some slight differences in format are observed between different developers' courses in the same team. The navigation structure of each course, the use of concept maps and the evaluation methods differ. Courses developed by the other countries involved in the project were also investigated. Some very significant differences were seen among the countries in the presentation of the content as well as the general structure.

For the EPPICC case there were no format differences among the modules since a formal structure was strictly imposed. All cases examined above were textual material with the addition of interactive examples, exercises etc. For the AVC-3 case, the structure of the courses was very different than each other, since video based lecturing was added. The presentation style of the videos was different. However, the major navigation structure was similar in all cases.

As a summary, the examination of material developed revealed that there are two basic approaches:

- Text based; with changing styles and formats
- Video based, again with changing styles.

These variations cause decrease in the quality of the courses in addition to the difficulties in assessment of the quality. Therefore, there is a need for common style guidelines which leaves some room to the developers based on the needs of their courses.

The DONC² Development Methodology

After the investigation of the all cases as well as the literature, the final version of the model was formed. A development model based on adaptive development approach which emphasizes iteration, concurrency, continuous feedback and collaboration was constructed. The necessary processes considered as components to develop courseware were determined. These components are positioned into three layers. Within the model, as emphasized by the circular structure used, there is no pre-determined sequence among the components, as depicted in Figure 6.



Figure 6. The DONC² Development Methodology

Iterations in the methodology are time-boxed phases, which generally take four to six weeks determined by project management, based on the amount of work assigned. All these components are assigned to different phases. This assignment does not emphasize any order as in traditional models. Any component can be assigned to a phase and can continue to be conducted in the following phases depending on the resources available. Management layer components (especially the planning activities) are conducted with a heavy workload at the very beginning of the project. Then integration layer components and individual courseware development activities in the micro layer start. All the components can be continued throughout the whole development process in different workloads. The approximate workloads of the components in phases can be seen in Figure 7. Regions with heavier loads are shown with wider black areas.

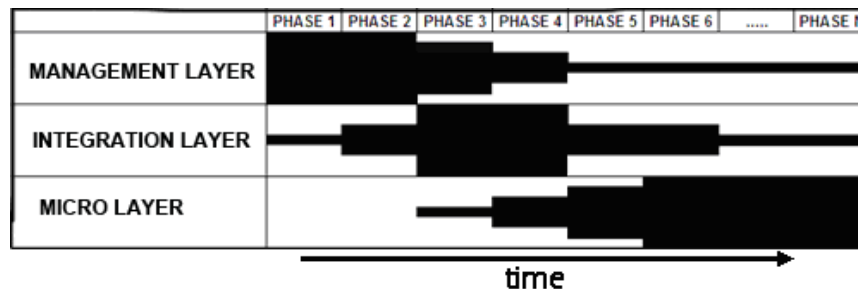


Figure 7. Workload change of three layers during the entire project

The complete workflow can be summarized in the form of phases and components as can be seen in Appendix A. At the beginning of each phase there is a phase planning meeting and at the end there are also review meetings for the evaluation and revision of what was done in that phase.

Management Layer Processes

The management layer deals with the strategic decision making activities of the project management team. Within the framework of management layer components, there is no pre-determined sequence. They all can be conducted concurrently as well as iteratively throughout the entire project. They can be divided to approximately four phases but this can be changed depending on the resources. These components involve tasks that are done throughout the whole development effort and they continue with different workloads till the end, as can be seen in Figure 8.

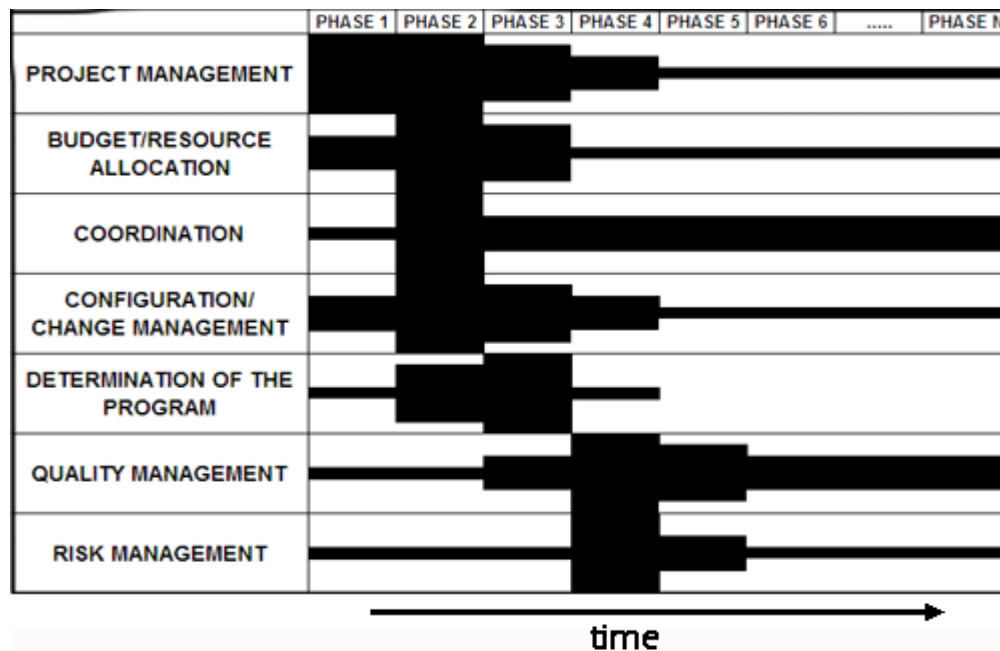


Figure 8. Workload change of management layer during the entire project

- *Project Management* is one of the main activities of the global layer which continues heavily until the end of the project and involves two major sub-activities:
 - Plan development activity is iterated several times throughout the whole process. It is revised based on the outputs of the other processes of the global layer. It does not make rigid projections for the later phases' components but rather gives outlines for the future tasks. Therefore planning is done at the end of each phase for the following phases in the form of phase planning meeting. This planning task also gathers feedback from the continuous revision and evaluation component which will be explained later.
 - Management starts from the beginning of the project and ends only when the project is finalized. It involves the control of the execution of the plans for the phases.
- *Budget/Resource Allocation* includes two major activities:
 - Budget/Resource Planning involves the determination of the resources in terms of human, equipment and materials and their quantities for the execution of the processes.
 - Organizational Planning mainly deals with human resource planning. This activity can be repeated during the project for several times according to the availability and workload of the people when performing macro or micro layer processes.
- *Determination of the curriculum* starts with the needs assessment activity. The workload in this component is heavier at the beginning while it is less at the later stages of the project but continues until the end as small modifications may be needed for the scope of the program.

- *Coordination* is done by the project management team to provide collaboration and communication among all the teams as well as team members.
 - Communication Planning: This involves determining the information and communications needs of the people involved in the project as well as information distribution mechanisms, which enable the required information available to people in a timely manner.
 - Communication Moderating is done by a coordinator or facilitator who is also the member of project management team.
- *Configuration\Change Management* is a component which is based on the emphasis on the change in the model. Continuous communication and feedback mechanisms exist among all layers as well as all components. This requires continuous revision and change for processes.
- *Quality control* is a required component as the quality of any degree program. Project management team determine the quality criteria as well as accreditation criteria for the degree program as well as the courses in the program
 - Quality Planning involves determination of quality policy, standards and regulations to be followed during the project and preparation of quality checklists for the evaluation of the developed curriculum and courseware.
 - Quality Inspections/Reviews is another continuous activity. This is also considered in continuous revision and evaluation component and this activity is conducted in the form of review meetings. These review meetings are conducted to check the quality of the components. In addition these will be in the form of usability evaluation meetings when they are especially conducted during the implementation of micro layer components.
- *Risk Management* is another essential component considered in global layer as this kind of development effort involves many risks that are to be overcome. Risks are generally based on hard or soft issues.
 - Risk Management planning involves deciding how to approach and plan the risk management activities for the project.
 - Risk Resolution: This activity is performed whenever any risk is occurred as a response to overcome the risk

Integration Layer Processes

The integration layer deals with the curriculum development activities. Integration layer components can be conducted in two phases iteratively and some activities of the processes continue with different workloads during the project as can be seen in Figure 9.

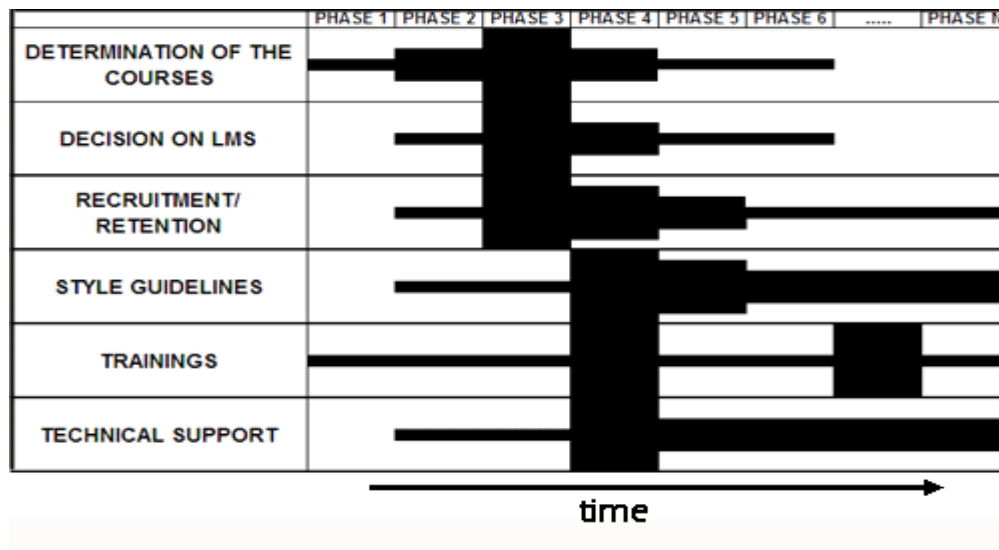


Figure 9. Workload change of integration layer during the entire project

- *Determination of the courses* component of macro layer aims to determine the courses to be included in the curriculum of the degree program. Courses and their scopes are identified according to the results of the need analysis activity based on the defined structure of the degree program.
- *Decision on LMS*, that will be used to offer the courses, is an important activity. The LMS that is easy to use and handled by the course developers as well as that have many interactive features for the students is to be selected. LMS is chosen after matching the requirements with the features of the investigated LMSs.
- *Recruitment/Retention* component is considered in macro layer as it is related with the target audience of the curriculum. It is about drawing attention of the target audience as well as keeping them on site. Consideration of this is necessary if the curriculum will be served for private education or trainings.
- *Style guidelines* provide standard structure in all courses according to the quality issues determined in the global layer. As these courses serve a degree program, commonalities are required in their style.
- *Training* is required for all people who work in any of these processes. Training programs are essential to give a common insight to distributed development teams on the requirements as well as the development strategies of online materials.
- *Technology support* deals with providing continuous support for development teams in their technological issues in a timely manner.

Micro Layer Processes

The micro layer deals with the courseware development activities. Micro layer components are carried out continuously until all the courses are developed as can be seen in Figure 10.

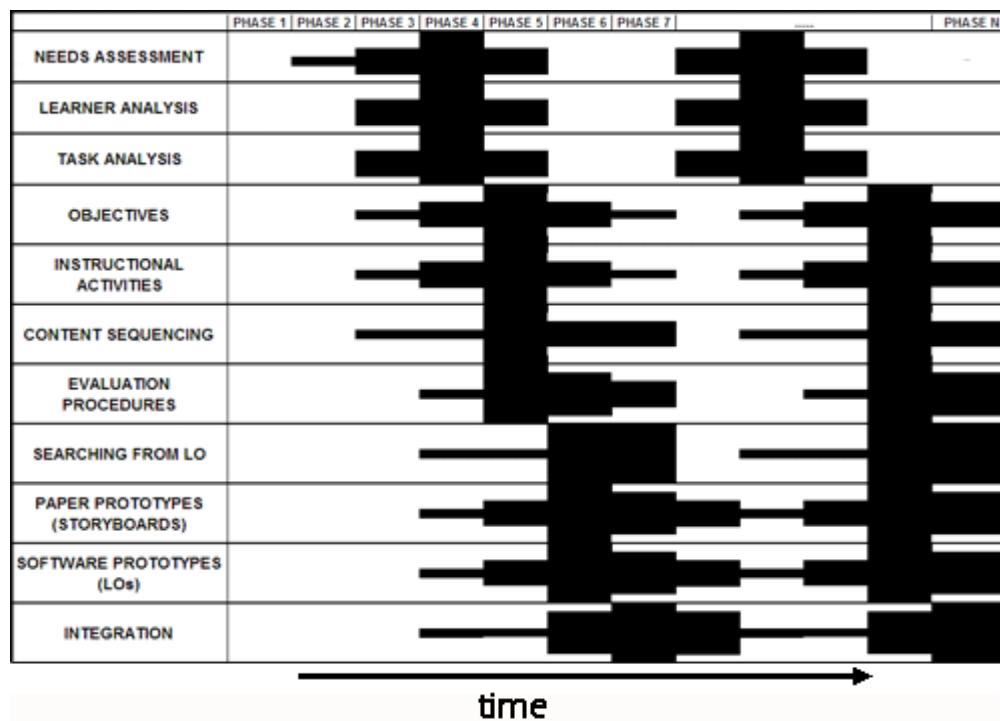


Figure 10. Workload change of micro layer during the entire project

- *Needs Assessment* component involves trying to gather what changes in students' knowledge are required by the developed course by an instructional designer.
- *Task Analysis* determines the content and the tasks necessary for the course. Subject experts are the primary source for this activity.
- *Learner Analysis* involves considering the target group's general characteristics, their prior knowledge as well as their motivation level and attitudes.

- *Determination of Goals\Objectives* enables the identification of the main aim of the course. They define what the learner would know or perform at the end of the instruction.
- *Instructional Activities* component involves determination of activities to be included in the learning environment to provide the interaction of the learners with the material, instructor and each other based on the determined goals and objectives.
- *Content Sequencing* involves the combination of the results of the task analysis, the goals and objectives of the instruction and instructional activities to decide on the sequence of the content of instruction.
- *Evaluation Procedures* that are going to be applied for the instruction are determined based on the determined goals and objectives.
- *Searching from learning objects* component deals with searching for suitable materials in existing learning object repositories before starting to develop new materials to reduce redundant efforts.
- *Paper prototypes (storyboards)* are the paper prototypes of learning materials. This will provide to evaluate the learning objects formatively by developing about 10 % of the course first to detect and reduce errors and developing the rest incrementally by supporting continuous evaluation.
- *Software prototypes (learning objects)* are the software prototypes of learning objects. All learning materials are developed as learning objects according to a determined standard. This will enable them to be re-used whenever necessary and reduce the redundant efforts. First 10 % of the paper prototypes are implemented as software prototypes than the rest is developed iteratively and incrementally.
- *Integration* involves the incorporation of developed learning objects according to the determined content sequence incrementally. They form a complete course in the degree program. All courses are also integrated and form a complete degree program

Communication

Communication is the essential and necessary process required for all level components and for all team members. Mechanisms are to be provided to enhance effective and efficient communication to ensure timely and appropriate generation, collection, dissemination, and storage of the project information especially for the geographically dispersed team members. Collaborative services that could be included in a communication mechanism can be listed as bulletin boards, discussion boards, e-mail, e-mail notifications, online paging/messaging, chat, white board, audio/video conferencing, task lists, contact management, screen sharing, surveys/polling, meeting minutes/records, meeting scheduling tools, presentation capability, project management, file and document sharing, document management, synchronous work on files/documents (Bafoutsou & Mentzas, 2002). These services are to be provided in a “shared workspace” (Poltrock & Engelbeck, 1997) environment in order to provide the critical links among people, ideas, and information necessary for success. This workspace functions as a shared memory as in the blackboard systems used in artificial intelligence (Corkill, 1991) which aim to find a solution to a common problem by different specialists. This way, each party can post a solution and apply their own expertise to any part of the problem and contribute to the overall solution. This can be provided through a web-based shared workspace infrastructure called communication blackboard. The infrastructure will contain specific areas in each layer of the model for different teams and team members can access several of these areas according to their responsibilities to contribute to others while solving problems. The communication process is continuously supported by the communication moderating activity of the coordination process of the global layer. A coordinator is responsible for the implementation of this component.

Evaluation\Revision

Continuous evaluation and revision are also essential elements and conducted at all layers. The processes at all layers are continuously tested and evaluated and revisions take place as a result of these evaluations. Evaluations can be either formative or summative. As a part of formative evaluation a series of usability tests are conducted. Revisions are conducted as peer reviews or expert reviews in any of the processes. This process is supported by the quality inspections/reviews activity of the quality control component of the global layer and takes place in the form of review meetings at the end of each phase. Based on the results feedback is given to the related components when necessary.

Personnel Requirements

Applying the DONC² methodology to an online curriculum and courseware development effort requires many roles which correspond to a different expertise. Subject Matter Expert (SME) is the specialist on the content of the course

- Project Manager (PM) is responsible for the entire implementation of the project and who has necessary managerial skills such as leading, communicating, negotiating.
- Team Managers (TM) are responsible for the work of each team at each level activity. They also should have the similar skills with project manager and they should serve and help to the PM for the implementation
- Coordinator/Facilitator (C) is responsible for the communication and enabling collaboration among all the team members. Whenever required, s/he should have the right to enforce even the project managers to enable the effective communication skills.
- Pedagogical Expert (PE) is responsible for the selection of pedagogical approaches that are going to be implemented in the courseware
- Technical Expert (TE) is responsible for the technical support of the development teams.
- Subject Matter Expert (SME) the specialist on the content of the course
- Visual Designer (VD) is mainly responsible for the style guideline defined in the project as well as the usability issues of the user-interface
- Multimedia Designer (MD) works in the design and development of multimedia materials used in the courseware.
- Audio/Video Director (AVD) is responsible for the creation as well as the preparation of video elements that will be used in the courseware especially in the video based lessons.
- Software Programmer (SP) is responsible for the development of software versions of the courseware in the form of learning objects with the integration of them as a complete courseware.

These roles can be matched with the components of the model and this can be seen in Appendix B. Marked intersections show the role which is primarily responsible in those components in the matrix. Some additional responsibilities can be given to these roles or they can act as feedback agents in other components.

Conclusion

Virtual learning environments (VLE), in other words, online learning environments, require features different from the traditional learning environments. This issue reveals the need for careful considerations in their design and development process as the online students deserve at least the same level of quality with their traditional counterparts. A distributed online curriculum and courseware development model (DONC²) was developed in this study to fulfill the aforementioned needs.

DONC² was developed by the investigation of related disciplines as well as the investigation of online courseware development projects. The related disciplines that some of the principles of the model were gathered were instructional design, software engineering, more specifically the adaptive software development (ASD) model. Four different online course material development cases were examined. The model was first formulated after the investigation of the first case. Then it was applied as an evaluation framework in one of the cases and it was applied as a development framework in another case. Based on the findings gathered from these cases, the essential principles for the model were revealed and integrated into the model.

As DONC² investigates the development effort components in different layers, they can all be adapted to the needs of the organization. It includes the important components that should be considered for the success of the development effort. It is a flexible model that does not attempt to impose rigid prescriptions to developers. It emphasizes short time-boxed iterations, collaboration at all levels among stakeholders, continuous evaluation and revision. Collaboration is essential for success and it can be achieved by effective communication which can be ensured by a coordinator who has effective communication and facilitating skills. Management based on leadership rather than an authoritative mode is emphasized for effective collaboration. Change is considered an indispensable feature so continuous updates are applied to plan.

The Model, while being improved, was applied in the final case (AVC-3) with success. Although it was not possible to put into practice all layers and components to the project, some of the components, especially, the supportive project management conducted with the help of coordination component, trainings provided to the developers and regular reviews conducted for the evaluation of the courseware were applied. The interviews conducted with the project members revealed that these components were considered as effective and helpful to the success of the development effort and quality of the work products.

As a future work, the whole model can be implemented fully to a complete program development case to investigate further if all determined components are working effectively. There is also a need to support and maintain the developed courseware to keep their quality. Therefore new courseware development cases can also be investigated to determine additional components related with the support and maintenance of the developed courseware or curriculum.

Acknowledgements

This research was supported by Human Resource Development Program of Turkish State Planning Agency

References

- Abrahamsson, P., Salo, O., Ronkainen J., & Warsta, J. (2002) *Agile software development methods: Review and Analysis*. Espoo, Finland: Technical Research Centre of Finland, VTT Publications 478, Retrieved June 26, 2006 from <http://www.vtt.fi/inf/pdf/publications/2002/P478.pdf>.
- Abran, A., Khelifi, A., Suryan, W., & Seffah, A. (2003). Usability Meanings and Interpretations in ISO Standards. *Software Quality Journal*, 11 (4), 325–338.
- Anido, L., Llamas, M., Fernández, M.J., Santos, J., & Caeiro, M. (2002). The Next Step in Computer Based Education: the Learning Technologies Standardisation. In Ortega, M. & Bravo, J. (Ed.) *Computers and Education*, pp. 71-80. Springer, Netherlands.
- Faulkner, C. (1998). *The essence of Human-Computer Interaction*. London, England: Prentice Hall.
- Anderson, J., Fleek, F., Garrity, K., & Drake, F. (2001). Integrating Usability Techniques into Software Development. *IEEE Software*, 18 (1), 46-53.
- Avicenna (2006). *Avicenna Project, Turkey home page*, Retrieved March 12, 2006 from <http://avicenna.ii.metu.edu.tr/>
- Bafoutsou, G., & Mentzas, G. (2002). Review and functional classification of collaborative systems. *International Journal of Information Management*, 22 (4), 281-305.
- Barjis, J. (2003). *An overview of virtual university studies: Issues, concepts and trends' from virtual education cases in learning & teaching technologies*, Retrieved December 21, 2004 from <http://www.idea-group.com/downloads/excerpts/2003/Albalooshi.pdf>
- Corkill, D. D. (1991). Blackboard Systems. *AI Expert*, 6 (9), 40-47.
- Crowther, M.S., Keller, C.K., & Waddoups, G.L. (2004). Improving the quality and effectiveness of computer-mediated instruction through usability evaluations. *British Journal of Educational Technology*, 35 (3), 289-303.
- Detweiler, M. (2007). Managing UCD within agile projects. *Interactions*, 14 (3), 40-42.
- Dongsong, Z.L., Zhao, J, Zhou, L., & Nunamaker, J. F. (2004). Can e-learning replace classroom learning? *Communications of the ACM*, 47 (5), 75-79.
- EPPICC (2006). *Equipping primary care physicians to improve care for children web site*, Retrieved September 12, 2006 from <http://www.eppicc.cme.uab.edu/>
- Ferre, X. (2003). Integration of Usability Techniques into the Software Development Process. In *Proceedings of International Conference on Software Engineering*, 3-4 May 2003 Portland, Oregon, USA

- Highsmith, J.A. (2000). *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. New York: Dorset House.
- Highsmith, J.A. (2002). What is Agile Software Development? *The Journal of Defense Software Engineering*, 15 (10), 4-9.
- Kemp, J.E., Morrison, G.R., & Ross, S.M. (2004). *Designing effective instruction*, 4th Edition, NY: John Wiley & Sons.
- Kruchten, P. (1998). *The Rational Unified Process: An Introduction*, Reading, Massachusetts, Addison Wesley Longman, Inc.
- Oliver, R., & McLoughlin, C. (1999). Curriculum and learning-resources issues arising from the use of web-based course support systems. *International Journal of Educational Telecommunications*, 5(4), 419-435.
- Poltrock, S.E., & Engelbeck, G. (1997). Requirements for a virtual collocation environment. *In Proceedings of the International ACM SIGGROUP conference on Supporting group work: The integration challenge*, 16-19 November 1997, Phoenix, Arizona, United States.
- Rtb (2007). *Rtb educational solutions web site*, Retrieved March 27, 2007 from <http://www.rtb.com.tr/>
- Ryan, S., Scott, B., Freeman, H., & Patel, D. (2000). *The virtual university: The internet and resource-based learning*. London, UK: Kogan Page Limited.
- Tripp, S.D., & Bichelmeyer, B. (1990). Rapid Prototyping: An alternative instructional design strategy. *Educational Technology Research and Development*, 38(1), 31-44.
- Willis, J. (1995). A recursive, reflective instructional design model based on constructivist-interpretivist theory. *Educational Technology*, 35 (6), 5-23.
- Xu, D., Wang, H., & Wang, M. (2005). A Conceptual Model of Personalized Virtual Learning Environments. *Expert Systems with Applications*, 29 (3) 525-534.
- Zhang, P., Carey, J., Te'eni, D., & Tremaine, M. (2005). Integrating Human-Computer Interaction development into the systems development life cycle: A methodology. *Communications of the Association for Information Systems*, 15 (1), 512-543.

Appendix A

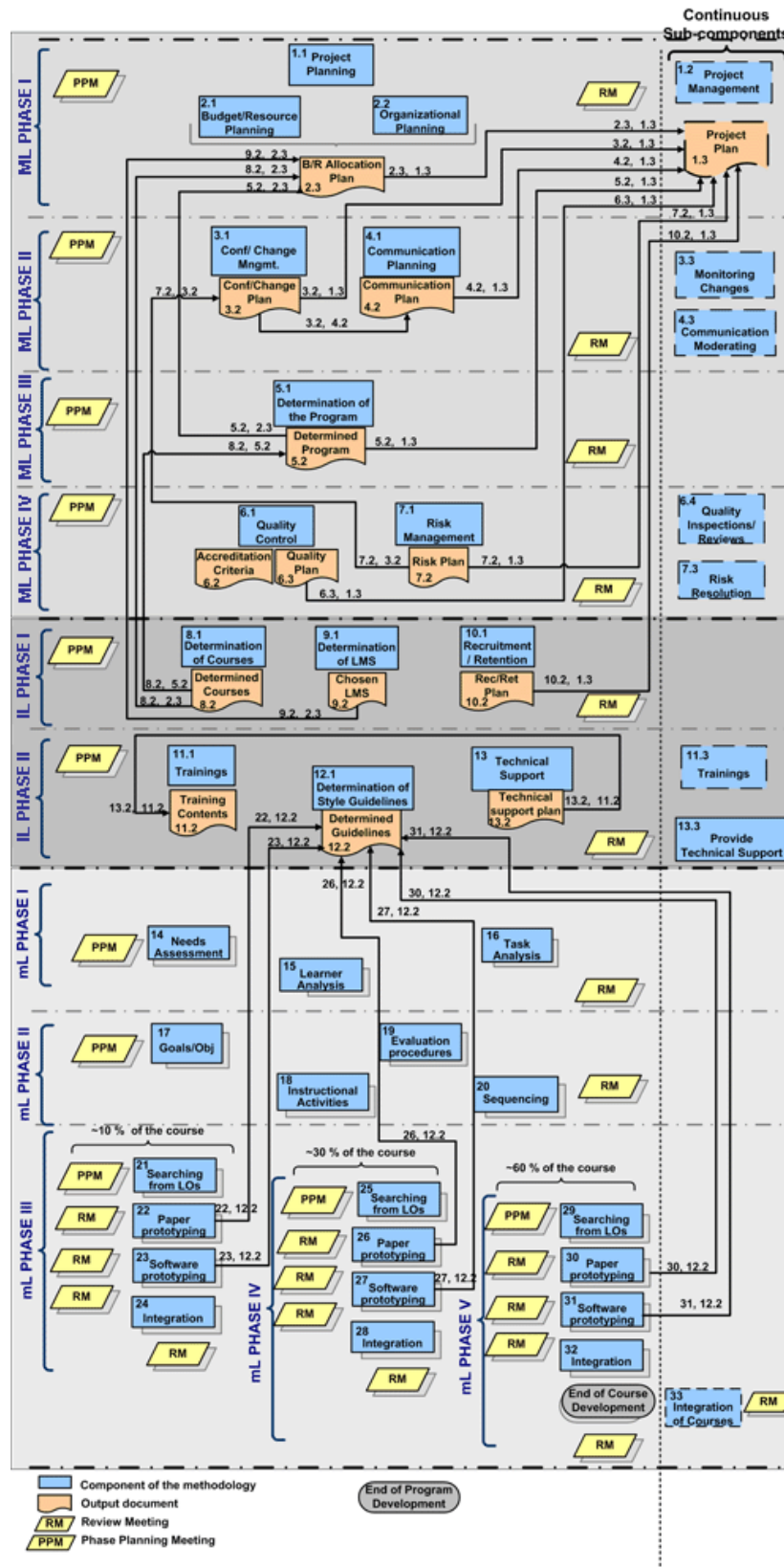


Figure A1. Overview of the methodology workflow

Appendix B

	PROCESSES / ROLES	PM	TM	C	TE	PE	SME	VD	MD	AVD	P
Management Layer Phase I	1. Project Management										
	1.1 Plan Development										
	1.2 Project Plan										
	1.3 Management										
	2. Budget/Resource Allocation										
	2.1 Budget/Resource Planning										
	Determination of resources										
	Cost estimation										
	Cost budgeting										
	2.2 Organizational Planning										
Staff Acquisition											
Team development											
2.3 Budget Resource Allocation Plan											
Management Layer Phase II	3.1 Configuration / Change Management										
	Determination of Change Control System										
	3.2 Configuration/Change Plan										
	3.3 Monitoring Changes										
	4. Coordination										
	4.1 Communication Planning										
	Determination of Information and Communication Needs										
	Determination of Performance Reporting Mechanisms										
	4.2 Communication Plan										
	4.3 Communication Moderating										
Man. Layer Phase III	5.1 Determination of the program										
	Needs analysis for the program										
	5.2 Determined Program										
Management Layer Phase IV	6.1 Quality Control										
	Determination of Accreditation criteria										
	Determination of Quality Issues										
	6.2 Accreditation Criteria										
	6.3 Quality Plan										
	6.4 Quality Inspections/ Reviews										
	7.1 Risk Management										
	Identification of Risks										
	Identification of Solutions										
	7.2 Risk Plan										
7.3 Risk Resolution											
Integration Layer Phase I	8.1 Determination of Courses										
	Needs analysis on courses										
	8.2 Determined Courses										
	9.1 Determination of LMS										
	Investigation of LMSs										
9.2 Chosen LMS											
Integration Layer Phase II	10.1 Recruitment/Retention										
	Identificaion of Rec/Ret strategies										
	10.2 Rec/Ret Plan										
Integration Layer Phase II	11.1 Trainings										
	Preparation of Trainings										
	11.2 Training contents										
	11.3 Continuous trainings										
	12.1 Determination of Style Guidelines										
	Examination of Style Guidelines										
	12.2 Determined guidelines										
13.1 Technology support											
Determination of responsibilities											
13.2 Technical support plan											
13.3 Technical support											
Micro Layer Phase I	14 Needs Assessment										
	15 Task Analysis										
Micro Layer Phase II	16 Learner Analysis										
	17 Determination of goals/objectives of courses										
	18 Determination of Instructional Activities										
Micro Layer Phase III	19 Determination of Evaluation Procedures										
	20 Sequencing the content										
	21 Searching from LOs										
Micro Layer Phase IV	22 Paper prototype preparation of 10 % of the course										
	23 Software prototype preparation of 10 % of the course										
	24 Integration										
Micro Layer Phase V	25 Searching from LOs										
	26 Paper prototype preparation of 30 % of the course										
	27 Software prototype preparation of 30 % of the course										
Micro Layer Phase V	28 Integration of 40 %										
	29 Searching from LOs										
	30 Paper prototype preparation of 60 % of the course										
Micro Layer Phase V	31 Software prototype preparation of 60 % of the course										
	32 Integration of 100 %										
	End of course development										
Micro Layer Phase V	33 Integration of Courses for the Program										
	End of program development										

Figure B1. Roles and Responsibilities matrix