CHAPTER 6

SCEnE USER AND BEHAVIOUR MODELING

6.1 Introduction

In the course of designing the prototype, the Unified Process was followed as an implementation methodology. Details of the prototype functionalities and software structures are described using the Unified Modeling Language (UML), through which the use case view and logical view of the prototype are modelled. The following sections present the details of SCEnE learning design specification and the UML-based modeling of the prototype.

6.2 SCEnE Learning Design Specification

The objective of the Learning Design Specification is to provide a containment framework of elements that can describe the design of a teaching-learning process in a formal way based on sound pedagogy and accommodating multiple intelligences theory. More specifically, the Learning Design Specification meets the following requirements.

Completeness: The specification is able to fully describe the teaching-learning process in a unit of learning, including references to the digital and non-digital learning objects and services needed during the process. This includes:

- integration of activities of both students and staff members (instructors, subject matter experts and administrators) integration of resources and services used during learning
• support for a wide variety of approaches to learning
• support for both single and multiple user models of learning
• support mixed mode (blended learning) as well as pure e-learning

**Pedagogical flexibility**: the specification is able to express the pedagogical meaning and functionality of the different data elements within the context of a unit of learning. It is flexible in the description of all different kinds of pedagogies and not prescribing any specific pedagogical approach.

**Personalisation**: The specification is able to describe personalization aspects within a learning design, so that the content and activities within a unit of learning can be adapted based on the preferences, portfolio, pre-knowledge, educational needs, and situational circumstances of users. In addition, the control over the adaptation process must be given, as described, to the student, a staff member, the computer, and/or the designer.

**Formalisation**: The specification describes a learning design in the context of a unit of learning in a formal way, so that automatic processing is possible.

**Reproducibility**: The specification describes the learning design abstracted in such a way that repeated executions in different settings with different person are possible.

**Interoperability**: The specification supports interoperability of learning designs.

**Reusability**: The specification makes it possible to identify, isolate, de-contextualise and exchange useful information artifacts, and to reuse these in other contexts.
The core concept of the SCEnE Learning Design Specification, as expressed in Figure 6.1, is that regardless of pedagogical approach, a person gets a role in the teaching-learning process, typically a student or an administrator role. In this role, he or she works towards certain outcomes by performing more or less structured learning and/or support activities within an environment. The environment consists of the appropriate learning objects and services to be used during the performance of the activities. Which role gets which activities at what moment in the process, is determined by the method or by a notification. The method is designed to meet learning objectives (specification of the outcomes for
students), and presupposes certain prerequisites (specification of the entry level for students).

Activities can be assembled into activity-structures. An activity-structure aggregates a set of related activities into a single structure, which can be associated to a role in a role-part. A structure can model a sequence or a selection of activities. In a sequence, a role has to complete the different activities in the structure in the order provided. In a selection, a role may select a given number of activities from the set provided in the activity-structure. This can, for instance, be used to model situations where students have to complete two activities, which they may freely select from a collection of activities contained in the activity structure.

6.2.1 Conceptual Terms of SCEnE Learning Design

In this section, an overview of the basic conceptual terms of the SCEnE Learning Design Specification is given.

*Unit of Learning*

A learning design is an integral part of any unit of learning. A ‘unit of learning’ is an abstract term used to refer to any delimited piece of education, such as a course, a module, a lesson, etc. It is noted that a ‘unit of learning’ represents more than just a collection of ordered resources to learn, it includes a variety of prescribed activities (problem solving activities, search activities, discussion activities, peer assessment activities, etc), assessments, services and support facilities provided by instructors and other staff members. Which activities, which resources, which roles and which workflow to adapt is dependent on the learning design in the units of learning.
Learning Design

A learning design is a description of a method enabling learners to attain certain learning objectives by performing certain learning activities in a certain order in the context of a certain learning environment. A learning design is based on the pedagogical principles of the designer and on specific domain and context variables (e.g. designs for mathematics teaching can differ from design for language teaching).

Learning Objectives

The learning objectives are the overall learning objectives to be attained by learners who complete the unit of learning. Learning objectives can be specified on several levels of detail. In SCEnE Learning Design, designers can choose to specify learning objectives at two levels, each with advantages and disadvantages. First, it is possible to define the learning objectives at the global level of the unit of learning. Second, it is possible to specify learning objectives for every single activity in the learning design emphasizing on multiple intelligences. Instructional designers can follow several approaches:

- Define the learning objectives only at the level of the unit of learning as a whole, not indicating the sub-objectives of the individual learning activities or what they add to the overall objectives.

- Define the learning objectives only per learning-activity and not globally for the unit of learning. The learning objective for the unit of learning is nothing more or less than the list of all the learning objectives specified in the different learning activities.

- Define the learning objectives on both levels: the learning objectives at the unit of learning level can be described more abstractly than those at the activity level.
Prerequisites

The prerequisites specify the overall entry requirements for students for doing the unit of learning. As with learning objectives, the prerequisites can be provided at the level of the unit of learning and/or for individual learning activities.

Roles

Roles allow the type of participant in a unit of learning to be specified. There are two basic Role types: Learner and Staff. These, however, can be sub-typed to allow learners to play different roles in certain types of learning activity such as task-based, role-play and simulations. Similarly support staff can be sub-typed and given more specialized roles, such as Tutor, Teaching Assistant, Mentor, etc.

The name a role is given depends on the pedagogy and setting used. In some instances a student is called a ‘learner’, in others, a ‘participant’. The names of staff roles are even more variant, e.g. instructor, trainer, facilitator, mentor, assessor, subject matter expert. Every role can have its own ‘title’ which provides the name for it. At runtime more than one user can be assigned to the same role, however restrictions can be set on the maximum and minimum number for each role. In this sense, roles are used for grouping purposes.

Activities

Activities are one of the core structural elements of the ‘learning workflow’ model for learning design. They form the link between the roles and the learning objects and services in the learning environment. They describe the activities a role has to undertake within a specified environment composed of learning objects and services. There are two basic types of activities: learning activities and support activities. A learning activity is directed at attaining a learning objective per individual user. Any user performs a learning activity only once (until completion). A support activity is meant to facilitate a role performing one
or more learning activities. More than one person can be assigned to a role in runtime. Activities can be aggregated into an activity-structure which provides the mechanisms to structure activities and referenced units of learning into a sequence or a user-selection.

Learning Activity: A Learning Activity consists of a single activity-description and several optional elements. The activity-description is the actual cue given to the user (rendered in the user-interface) to describe the activity to be performed by the user. In most cases, the activity-description is a text (of type webcontent). In other cases, it can be an image, audio-file (webcontent), a video file or any other cue to the user.

Support Activity: A Support Activity consists mostly of the same elements as a Learning Activity, but without the learning objectives, prerequisites. Example: a staff role has the support activity to grade reports made by persons in the learner role named ‘student’. Every person being a student creates his/her own report. The tutor grades every report.

Environment

Activities take place in a so-called ‘environment’, which is a structured collection of learning objects, services, and sub-environments. The relationship between an activity and an environment can be derived from the linguistic description of the activities. Most nouns in the activity imply the availability of learning objects in the environment; references to other persons imply the availability of communication services; some verbs imply the availability of supportive services or tools. For instance the activity: ‘read the case study and discuss solutions with your peers’ refers to environment components: ‘the case study’ which must be available for reading; and ‘peers’ which must be available to communicate with (including communication means).

Learning Object

Learning objects are defined here, as any reproducible and addressable digital or non-digital resource used to perform learning activities or support activities. Examples are: web

**Service**

Besides resources which can be defined at design time, there are numerous so-called ‘service facilities’ used during the teaching and learning, for instance, a discussion forum or some other communication facility. Service facilities are resources that cannot be given a URL at design time. They have to be instantiated by a local runtime service. This is because, if a service facility is bound at design time, then that specific service would have to be used by all users of all instances of the learning design. When what is needed is an instance of the service that is unique to the runtime instance of the learning design and its assigned users, (e.g., if a chat forum is to be dedicated to the use of a specific group of students and support staff associated with a particular instance of a learning design), then this has to be created and the local URL assigned after the instance of the design has been set up and the group of students and staff associated with it. For this to work, it requires a well defined set of service type, which are known to the runtime service, such as chat, discussion forum, bulletin board and e-mail.

**Conditions**

Conditions are used in conjunction with properties to further refinement and to add personalization facilities in the learning design. Conditions have the basic format: IF [expression] THEN [show, hide, or change something or notify someone].

**Notification**

With notifications it is possible to send a message to a role or to assign new learning or support activities to roles based on certain events. The events include the completion of a certain activity, the completion of the unit of learning and when an expression in a certain condition is true.
6.3 Use Case View

This section presents the use case diagram that depicts the user centric view of the prototype’s functionalities. This includes the different learning activities in an e-learning environment that seeks to accommodate several differing pedagogical approaches in order to cater for different learning styles based on Multiple Intelligences (MI) theory.

For the purpose of making the derived use cases as simple as possible, the researcher grouped the different use cases based on views of the different types of prototype users. This includes the subject matter expert’s view, instructional designer’s view, instructor’s view, admin-staff’s view and student’s view as a generic view.

The staff’s view only describes the use cases initiated by the respective type of staff. They are related to the tasks restricted to the administrator. The restricted access of these use cases is imposed as away of controlling the consistency of the resultant e-learning environment. Staff can be sub-typed and given more specialised roles, such as Subject Matter Expert (SME), instructional designer, instructor, tutor, teaching assistant, mentor, admin-staff, etc.

6.3.1 Instructional Designer’s View

The instructional designer’s view only describes the use cases initiated by the instructional designer. They are related to the tasks restricted to the instructional designer. The restricted access of these use cases is imposed as a way of derivation of unit of learning based on sound pedagogy and adhering to multiple intelligences theory to certain extent. Justification of this approach is described in (Krishnasamy et al., 2007). Figure 6.2 shows the use cases associated to the instructional designer’s view.
6.3.2 Subject Matter Expert’s View

The subject matter expert’s view only describes the use cases initiated by subject matter experts (SME). This includes the management of units of learning of a particular course and issues related to the outline of a topic and the measurement of subtlety. Figure 6.3 shows the use cases associated to the subject matter experts.
6.3.3 Instructor’s View

The instructor’s view contains tasks related to staff that teaches the course or assigned to develop the course. This comprises of creating content in variety of formats, prepare examples, building exercise, creating test bank, identifying appropriate activities to support the intelligence type and also determining the dominant and supporting intelligence of their students, if the instructor is teaching the course. The diagram shown in Figure 6.4 depicts uses cases that are associated to the instructor’s view.
6.3.4 Admin-staff’s View

The admin-staff view is a generic one, any worker belonging to the learning organization is considered as an admin-staff. Hence, this view represents the use cases through which all the basic functionalities of the prototype are realised. Figure 6.5 illustrates the use cases that are associated to the admin-staff’s view.
6.3.5 Student’s View

The student’s view is meant to provide the students of the learning organization the ability to browse and actively involve in the e-learning process that accommodates multiple intelligences. The diagram as shown in Figure 6.6 depicts use cases that related to the student’s view.
6.4 Use Case Realisation

The use case realisation includes a description of the classes that are involved of carrying out each individual use case. It also involves the description of the different types of interactions between the use case elements. Some of the use cases shown above are very simple and they do not carry any complex tasks. Thus, only use cases that perform complex tasks are described. This is in line with the purpose of the system modeling approach where only complex data or processes should have the focus in the modeling process.
6.4.1 Establish Student Information

The establish student information (maintain student) use case is part of the admin-staff’s view. Its purpose is to add, update or delete a student as part of the e-learning community. Student information also includes the preferred type of intelligence of the student and the intelligence type that need to be improved. For example a student can be dominant in verbal-linguistic intelligence but poor in interpersonal intelligence. Figure 6.7 shows this use case’s interaction diagram in the form of UML sequence diagram.

Figure 6.7: Sequence diagram of the use case: Establish student information
6.4.2 Approve Member’s Access

The approve member access use case is also part of the view of the admin-staff. Its purpose is to approve or reject members’ application to subscribe in respective e-learning communities. Figure 6.8 illustrates this use case’s interaction diagram in the form of UML sequence diagram.

![Sequence diagram of the use case: Approve member’s access – basic flow]

Figure 6.8: Sequence diagram of the use case: Approve member’s access – basic flow

6.4.3 Login – Basic Flow

The login use case is also part of the generic view of admin-staff. It enables an admin-staff to update the login capabilities of members based on their role. Figure 6.9 shows the basic flow of login use case interaction diagram in the form of UML’s sequence diagram. Other
interaction diagrams related to this use case are login cancel, login incorrect and database connection error.

![Sequence diagram of the use case: Login – basic flow]

**Figure 6.9:** Sequence diagram of the use case: Login – basic flow

### 6.4.4 Maintain Module

The maintain module use case is part of the generic view of instructional designer’s. It enables an instructional designer to add, edit or delete module details. This includes the module breakdown, referred to as unit of learning and the information pertaining to this unit of module. The task is done only once to prevent adding redundant module with different code or title. Instructional designers can later add or edit the module details.
through another view (i.e. edit module). Figure 6.10 illustrates the basic flow of maintain module use case interaction diagram in the form of UML’s sequence diagram.

![Sequence diagram of the use case: Maintain module](image)

Figure 6.10: Sequence diagram of the use case: Maintain module

### 6.4.5 Course Work Maintenance – Basic Flow

Course work maintenance is carried by grade student use case and is part of the instructors. Its purpose is to enable instructors to grade students based on different types of assessment approaches supporting multiple intelligences. The assessment approaches include individual or group assignments/projects, class activities or field work and online participation. Figure 6.11 depicts this use case’s interaction diagram in the form of UML sequence diagram.
6.4.6 Track Progress

Course work maintenance is carried by grade student use case and is part of the instructors and admin-staff views. Its purpose is to enable instructors to grade students based on
different types of assessment approaches and to keep track of their progress. On the other hand, admin-staff calculates the GPA and CGPA of the students based on the grades given by the instructors for their final exam. Figure 6.12 depicts this use case’s interaction diagram in the form of UML sequence diagram.
Figure 6.12: Sequence diagram of the use case: Track progress
6.4.7 Set Learning Objectives

The set learning objectives use case is part of the subject matter expert’s view. Each learning objective relates to one or more learning outcome. Figure 6.13 shows this use case’s interaction diagram in the form of UML sequence diagram.

Figure 6.13: Sequence diagram of the use case: Set learning objectives
6.4.8 Determine Prerequisites

The determine prerequisite is also part of the subject matter experts view. Its purpose is to allow students to construct knowledge by making sense of experience in terms of what is already known. This use case’s interaction diagram in the form of UML sequence diagram is shown in Figure 6.14.

*Figure 6.14:* Sequence diagram of the use case: Determine prerequisite
6.4.9 Create Topic Content

The create topic content which is under the instructors’ view, enables respective instructors to create topic contents for a module. There are varieties of presentation styles of the contents. This includes narrative, audio, video, graphic, spreadsheet format and even presented as a game such as puzzle. This is in accordance with multiple intelligences theory to cater for diverse preferences. Figure 6.15 illustrates the basic flow of this use case’s interaction diagram in the form of sequence diagram.

Figure 6.15: Sequence diagram of the use case: Create topic content
6.4.10 Create Example

The create example use case is part of the generic view of instructors. It enables instructors to provide sample questions and answers for the relevant topic. The questions and answers can be presented in different ways. For example, multiple choices, true/false, short answers and essays. Figure 6.16 shows this use case’s interaction diagram in the form of UML’s sequence diagram.

Figure 6.16: Sequence diagram of the use case: Create example
6.4.11 Build Exercise

The build exercise use case is also part of the generic view of instructors. The tasks related to this use case are building various types of exercises as well as providing feedback link, feedback message and brief explanation on the area questioned. The feedback enables student to attempt the question firstly and get the proper answer even if they have not mastered the topic well. Figure 6.17 depicts this use case’s interaction diagram in the form of UML’s sequence diagram.
6.4.12 Prepare Assignment/Project

The prepare assignment use case enables respective instructors to prepare assignments or project papers based on the project type and category. Project type refers to individual or group based work where else project category refers to different classification of assignment/project to support multiple intelligences. The categories are field work, experiment, portfolios, interview, online collaboration and research. Figure 6.18 shows the interaction diagram for this use case in the form of UML’s sequence diagram.

Figure 6.18: Sequence diagram of the use case: Prepare project
6.4.13 Load Glossary

The load glossary use case includes other use cases such as add, edit or delete terms. This use case is handled by the subject matter expert. A subject matter expert inserts a term into a glossary database for any unit of learning created. As they create more units of learning, the expanded glossary is available to all other modules. Glossary definitions may also include image. Figure 6.19 shows the basic flow of this use case’s interaction diagram in the form of UML’s sequence diagram.

*Figure 6.19: Sequence diagram of the use case: Load glossary*
6.4.14 Maintain Summary

The maintain summary use case is extended to add auxiliary link, specify learning outcome and identify next steps. These use cases are part of the subject matter experts' view. Its purpose is to provide auxiliary links related to the topic, outline learning outcome to sum up what have been covered in the topic and also show students the next topics to be looked into. Figure 6.20 shows the basic flow of this use case’s interaction diagram in the form of UML’s sequence diagram.

![Sequence diagram of the use case: Maintain summary](image)

Figure 6.20: Sequence diagram of the use case: Maintain summary
6.5 SCEnE Logical View

This part presents the structures that comprise the skeleton of the system being modeled. The class diagram acts as a data dictionary that describes all possible data structures and relationships that control the input and output of the system’s data. On the other hand, the state chart diagram is used to describe all possible states of the activities sequence which represents the elementary features and functions of SCEnE approach.

6.5.1 SCEnE Class Diagram

The UML class diagram is used to describe the classes involved in realising the functionalities on hand. The diagram also shows the interrelationship between candidate classes. Figure 6.21 presents the class diagram of the proposed e-learning approach. The diagram is based on the assumption that staff (instructor, instructional designer, subject matter expert and admin staff) play an important role in creating the e-learning environment. Staff handles the modules. The module here refers to a course or subject (e.g. Science). Each module contains few chapters and each chapter is further split up into topics.
As explained in Section 3.5, limited content presented in a topic makes it easier for students to complete in one sitting and it forces the instructor to concentrate on a few main concepts in that topic. This approach also fits ideally with the concept of a learning cycle. A learning cycle in this context refers to knowing the learning objectives and prerequisites (if any), understanding lesson content which is presented in multiple formats, going through some examples, doing some exercises, participate in assignments or projects and finally recap by looking into the summary. In order to support learning activities SCEnE provides supporting activities which comprise of e-mail, forum, chat, and homework drop box.
6.5.2 Activity Diagram

Since the entity lesson content is representing the basic building block of the proposed e-learning approach, modeling the actions and object flow of this entity is very important to understand this entity’s behaviour. Lesson content activity diagram illustrated in Figure 6.22 is a dynamic diagram that shows the activity and the event that causes the content object to be in a particular state. This activity diagram is a simple way to represent the workflows and their steps of SCEnE system. As described in Chapter Three, the quality of e-learning is not determined individually by the topic content. The state of a topic content changes based on the learning outcome, relevant examples, exercises, projects, test bank and appropriate feedback. This lesson content creation process is carried out by members of e-learning communities that hold respective roles.
Figure 6.22: Lesson Content Activity Diagram
6.6 Summary

In this chapter the researcher has described SCEnE learning design specification and the UML-based modeling of the prototype. The UML based modeling were on use cases and sequence diagram for the selected use cases, and the logical view which includes SCEnE class diagram and lesson content activity diagram. With reference to this SCEnE software modeling, the prototype implementation of SCEnE is discussed in the following chapter.