CHAPTER 4

FRAMEWORK OF ONLINE ASSESSMENT MANAGEMENT SYSTEM

This chapter deals with the general framework of online assessment management system (OAMS) for online formative knowledge assessment and the basic building blocks involved in this framework.

A framework is a basic conceptual structure used to solve complex issues. It consists of a set of tools, materials or components or objects. A framework ties all the elements involved in a particular process. In a software context the word framework is used as a term that represents different kind of toolsets and component bases. Generally, it serves as a guideline to implement the identified tasks. The proposed work demands a web based application that involves various elements to perform this research work.

Keeping the objectives of this research work, a framework was developed for implementing the proposed methods. The general framework of the online assessment management system (OAMS) is depicted in Figure 4.1.

4.1 DEVELOPMENT OF OAMS

The following elements were considered for inclusion into the framework: the database server, the client application and the application server.
Figure 4.1 Framework of Online Assessment Management System (OAMS)

The framework was implemented by using the following: Eclipse Juno, Apache Tomcat, MySQL, Java Server Pages (JSP) and Java Scripts. Apache was chosen as web server which is a container of servlets. A servlet is a small Java interface, which runs within a web server. Servlets receive clients’ requests and respond to them across Hyper Text Transfer Protocol. It is used to extend the capabilities of servers that host applications accessed via a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by Web servers.

Eclipse Juno is a familiar IDE for java based web applications. Juno is based on the Eclipse 4.x platform. Eclipse Juno includes a compatibility layer that allows existing Eclipse plug-ins and Rich Client Platform (RCP) applications to work on the new platform. To deploy and run a Servlet, a Web container must be used. A Web container is a component of a Web server that interacts with the servlets. It is responsible for managing the
lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights. Apache Tomcat is an open source web server and servlet container developed by the Apache Software Foundation (ASF). Apache also supports PHP and MySQL for web applications to be hosted.

MySQL has been popular because of many good reasons. MySQL is an open-source RDBMS software and a very powerful program which can handle a large subset of the functionality of the most expensive and powerful database packages. It can also work on many operating systems and with many languages. More importantly MySQL is customizable. The open source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

The application on the client side implements the conceptual layers i.e. client functionalities, CmapEditor and other components of the user interface. The user’s actions in the user interface like pressing of buttons, entering text into a text field, viewing assessments, editing profile, creating maps and others are examples for events. The client application handles different events and calls services of the application server. Called services are executed remotely on the application server, which acts as a controller.

The web server receives remote calls from the client and redirects them to the appropriate servlet. The information about a servlet is included in the remote call. The servlet handles a call and launches the appropriate method needed for the communication with the database or for the execution of the event. The server does not use MySQL queries to perform data manipulations. The application is secure enough because the database server is opened for connections only from the server and this will not allow any other person from outside to access the data directly. The system is designed
to provide only the asynchronous communication through its email services feature. A screen shot of the OAMS home page is presented in Figure 4.2.

![OAMS Home Page](image)

**Figure 4.2 OAMS home page**

OAMS provides interface to create source concept maps and store it into the database. The educator login with his user name and password then chooses the type of assessment and creates the source map using the JavaScript based interface. The concept maps for the proposed OFKA were designed based on the content from textbook of the courses. The topics covered in classes conducted during a specific week were considered for the assessment and maps were created accordingly.
4.1.1 Add Assessment Process

OAMS was developed based on concept maps (CMs), which were used for two assessment approaches: CCMG and CTG. The topics chosen for the assessments were taught in the classes in the preceding conduct hours. The educator decides the topic for the assessment then creates the assessment using the add assessment option from menu. The announcement of the assessment include the following details; assessment topic, description, instruction, start date, end date and type of assessment. The source map for the assessment is created by the educator using the CmapEditor which stores the concepts and relations into the database as JSON structure (JavaScript Object Notation). A screen shot of the add assessment page is presented in Figure 4.3.

Figure 4.3 Screenshot of Add Assessment
4.2 BUILDING BLOCKS OF ONLINE ASSESSMENT MANAGEMENT SYSTEM

The success of any OFKA system relies on process of selection and definition of the basic building blocks. OAMS was designed to have the following elements as its basic building blocks; a user interface to access basic services of OAMS, a user friendly map editor called CmapEditor, a game engine that takes care of the proposed games using the game rules defined, a feedback generator responsible for provision of feedback and a report generator for reports.

4.2.1 CmapEditor

CmapEditor is an important element of the OAMS which provides the basic functionalities needed to create and manage concept maps for the assessments. The functionalities include add node, add relationship, change node name, change relation name, remove node, remove relation and save. Many existing map editors do not provide any interactive feature like hiding the nodes when needed. But CmapEditor is designed to provide this feature. For example, if the map contains three child nodes under the root node and if the user is navigating through the first child node, then the child nodes of other two nodes will be hidden. This enables the user to have a comfortable view of the nodes and links of the maps. A screen shot of the CmapEditor is presented in Figure 4.4.
The CmapEditor is developed using JSP, JSON and MySQL technologies. CmapEditor enables the user to create CM for desired topic. The nodes and relationship labels are stored as JSON data structure into the MySQL database. Further, the editor has been designed to have the advanced user interface techniques like drag and drop of objects in the canvas. Major functionalities along with other menu based features are implemented using the JavaScript.

4.2.2 Game Engine

The Game Engine (GE) is one of the core elements of the OAMS as it was designed to handle the game procedures. In general, a GE is the part of the game that draws all the pictures, handles game events, deals with communication, saving games and details to database, etc. In the context of this research work, the proposed OFKA games were designed as simple
games and thus the GE was designed to have simple functionalities. The GE was designed to perform actions related to the proposed CM based games, the CCMG and CTG. A detailed discussion on the development of proposed algorithms and other procedure adapted are presented in the next chapter.

4.2.3 Feedback Generation

Another basic building block of OAMS is the feedback generation process. As discussed earlier, the role of feedback in formative knowledge assessment techniques is regarded as highly important and crucial. Hence, the researchers took more care to design and develop the feedback mechanism. The present feedback generation method is a simple approach, which follows a direct matching approach on the source maps and resultant map during the OFKA methods.

4.2.3.1 Feedback generation for students

In both OFKA assessment methods, the GE validates the student’s map with the source map (Educator’s map) at the end of the game. The nodes and relations are compared one to one. The validity of the moves will get the students the defined scores. Further, after the evaluation of the resultant map, the GE analyzes the invalid nodes and relations based on the proposition statements formulated from the source map. Based on the analysis, the feedback is generated and displayed to the students immediately along with the scores.

The feedback consists of the invalid concepts identified from the analysis. A simple statement containing the invalid concepts make the student to realize the misconceptions on the concepts of the topic and the answer is also provided to the student to correct the misconceptions instantly. Thus the feedback aids the student towards better learning.
4.2.3.2 Feedback generation for educators

The system stores the feedback generated for all the students into the database. Once the deadline for the assessment is reached, the system closes the assessment and then invokes the feedback system to prepare a consolidated feedback to the educator. All the students’ results will be analyzed using simple grouping mechanism, which groups the mistakes and then ranks them. Based on the results of the analysis, a feedback on the performance and misconceptions of the students provided to the educator. Then the educator performs the remedial actions for that assessment. The mechanism followed for the remedial actions is shown in Figure 4.5.
4.2.4 **Report Generator**

The report generator is another important element of the OAMS, which is responsible for the generation of reports that are needed by the educators. Normally the reports in terms of participation in the assessment, performance of the students’, assessments class wise and subject wise can be generated using the report generator. This helps the educator to maintain a record of the assessment activities for future references.

4.2.5 **Collapsed Concept Map Game**

The Collapsed Concept Map Game (CCMG) is one of the proposed OFKA methods. This is simple concept game in which the concepts are presented in shuffled manner to gauge the students’ understanding. For the CCMG approach a simple gaming procedure is deployed into the game engine which takes care of the game functionalities tasks. More detailed discussion on the development of CCMG method is presented in the next chapter.

4.2.6 **Concept Tree Game**

The Concept Tree Game is another OFKA method provided by the OAMS. The CTG follows simple concept tree construction game. The game has two players; the system and the student. The allowed actions in the game are addnode, addrelation, skip and quit. The system and student will construct the tree using the concepts and relations in appropriate place alternatively as a game. This kind of game based assessments with fun and excitement motivates students towards taking assessments without any hesitation. The game follows the extensive game theory in analyzing and awarding payoff for the players’ decision on next move. The game has two scores; validity score and payoff. Every choice of action decided by the
student will get a predefined point and every valid concept or relation will also get a point to the student. This method also provides an immediate feedback to the students based on their performance in the assessment. More detailed discussion on the development of CTG method is presented in the next chapter.

4.3 CHAPTER SUMMARY

The development of OAMS was discussed earlier in this chapter. The general framework was presented along with discussion of its basic building blocks. The procedure and trial of the proposed methods are dealt in the next chapter.