The Architectural Design of an Integrated Virtual Classroom System

A.I. Obasa, A.A. Eludire and Mbing Isaac

1Department of Mathematics, Statistics and Computer Science, Kaduna Polytechnic, Kaduna, Nigeria
2Department of Computer Science, Joseph Ayo Babalola University, Ileje Arakeji, Osun State, Nigeria

Abstract: This study discusses the architectural design of an integrated system for the delivery of lectures in a virtual environment. The architecture and description of the system components are presented with the techniques and recommendations for the implementation of the designed system. The system architecture is multi-tier, modular, scalable and built for adaptability to database middleware suite. All functionalities within the application are delivered using Web Services, communicate via industry standard XML messaging and access is purely via a web browser. The virtual classroom application utilizes an operating system with facilities for providing users web services, data and pre-processor hypertext with a simple and consistent User Interface corresponding to the tiers. With the object-oriented design method used in this application, three logical categories were identified for application objects, namely, objects that deal with presentation aspects of the application, objects that deal with the business rules and data, and objects that accept and interpret user requests and control the business objects to fulfil these requests.

Key words: Application objects, client-tier, multiple-tier, user interface, web-tier

INTRODUCTION

Collaborative learning: (Collis, 1996; McNabb, 1994) stated that successful distance education systems involve interactivity between teacher and students, between students and the learning environment, and among students themselves, as well as active learning in the classroom. Hiltz (1994) stated that collaboration and feedback from other students make virtual classroom to overcome the absence of face-to-face communication. Most of the existing virtual classroom platforms lack collaborative learning facility, which is an integral component of teaching, and learning system.

Limited interaction: A good number of available teaching and learning platforms use facilities like e-mail, wikis and bulletin board to provide students interaction. This approach is grossly inadequate as the teacher may not be able to reply all mails that may be received and where received mails are replied the reply may not be immediate. This process limits the amount and depth of interactions regarding course materials and procedures. It also removes any feelings of connection between the student and instructor.

Lack of two-way real-time communication with faculties: A good number of existing virtual classroom platforms are asynchronous and do not facilitate two-way real-time communication (Eluminate, 2007, 2008). There is need to incorporate many interactive features, for both on-line and off-line use to facilitate the two-way communication between students and faculties, thereby alleviating the lack of sense of belonging of remote students.

One of the solutions to the above is to integrate asynchronous and synchronous learning platforms for virtual teaching in higher Institution of learning. This solution requires an architecture that can resolve some of the limitations noted in achieving collaborative learning. The study therefore, has the objective of proposing an architectural design that foster the solution to the limitations posed to collaborative learning, interactions amongst learners, lack of two-way communication with faculties and learner’s interactions.

The proposed architecture comprises of four interrelated modules, namely, Application tiers, Application objects, Data Processing module and Course module (Fig. 1) realised in multi-tier levels. The implementation of this architecture over web-based services has substantially resolved issues with the identified limitations.

APPLICATION TIERS

The integrated virtual classroom system is a multiple-tiers application consisting of Client-tier, (CT), Web-tier
(WT) and Enterprise Information System-tier (EIS) as shown in Fig. 2. A tier can be anywhere in the network, local or remote; can be implemented with any programming languages.

The client-tier application provides a Graphic User Interface (GUI) to the end users, which is achieved using a web browser, such as Internet Explorer, Opera or Firefox. The HTML and PHP with Java applet is used to build the client-side GUI page. A web server, Apache or Internet Information Server (IIS), which is bundled with Microsoft 's Office server reference implementation, is used to serve the GUI page at the web-tier, and to provide dynamic content HTML/PHP pages. The web-tier holds data models of the business data, presents the data models to the client through HTML/PHP pages, accepts and analyzes the user's inputs, passes the user's request to the Enterprise tier for processing, and forward response back to the client. The WT handles the business logic and data. It communicates with the Enterprise Information System tier through appropriate protocols. A relational database MySQL version 5.2 and PHP are used to build the EIS tier application.

The database should be of a powerful DBMS technology that can handle application security. The MySQL's Process Manager records and provides a viewable audit of all message transactions between web services within the system. The front end is delivered as pure HTML to ensure that it is extremely lightweight and browser independent. The web-tier application communicates with the EIS-tier application through a
APPLICATION OBJECTS

The object-oriented design method is used in this application. The process of object-oriented design begins with the identification of business objects, data, and presentation objects and model each of them in software. At the highest level, the application objects can be divided into three logical categories (Fig. 3):

- objects that deal with presentation aspects of the application,
- objects that deal with the business rules and data, and
- objects that accept and interpret user requests and control the business objects to fulfil these requests.

Then the data objects are mapped to the business objects, then business objects to presentation objects.

DATA PROCESSING MODULE

One of the first decisions taken in this project concerns the tiers that the application uses. The chosen platform is designed for multi-tier applications, and offers a lot of flexibility in choosing how to distribute application functionality across the tiers. In a Web-enabled application, such as this application, some tiers are always present. There is the client tier, that is, the browser. The browser communicates with the Web tier. Finally, the EIS or database tier holds persistent application data. However, there is a choice in whether the Web tier accesses the EIS resources directly, or goes through an Enterprise Java Bean (EJB) tier. The decision depends on the functionality, complexity, and scalability requirements of the application in the real world. Since such requirements can change as the application evolves, one goal for the design is to make it amenable to migration to an EJB-centric approach therefore, this application is a multi-tier EJB-centric application. Figure 4 shows the architecture of an integrated multi-tier application. It illustrates the flow of data through the application.

USABILITY SPECIFICATION

This application models a web-based student course delivery system. The application interface is presented to the students through a hypothetic web site (http://localhost/students). The student interacts with the application using a web browser. The application presents the student with a catalog of courses from which the student selects course offerings and register for the selected courses. New students are required to register using a combination of username and password. The system notifies the student of course registration by a confirmation email using JavaMail API. The student can always make a query to see what courses he/she already registered earlier. When the student is ready, then, proceed to attend the lecture delivery either in asynchronous or synchronous mode.
The usability specification is the key ways the application would be used. The kinds of functionality which are available from the user interface of the application are:

- A common set of links or navigation bars on each page that provide the student with quick access to common navigational tasks.
- An organized view of the site’s contents through a categorized course catalog. A search mechanism to provide students a way to locate courses based on key-word descriptions.
- A master view of the catalog that lists course offerings of interest to the student. This could be the result of the user navigating through a catalog category or the outcome of a keyword searches.
- A detail view that describes the details of a particular course offering. Students can click on an offering in the master view to zoom in on the details of that particular course.
- A chatting facility for student registered on a particular course.
- Access to a forum where users (students and teachers) can exchange ideas on different issues relating to a given course.
- Access to submission and uploading of assignment and subsequent access to the graded result.

**Logical flow:** A student connects to the application. If the student logs in, the user module maintains student's course registration information. It creates new student records and manages these records. Student records include such information as student ID, name, password, address, phone and email address:

- The course module returns a list of courses available. The course catalog module searches the course database for a list of possible matches to the search criteria and renders the courses for the student user.
- The student views a specific course. The course catalog module also returns detail information about the selected course.
- The student selects a course offering for registration. The registering client module creates a registering cart for the student for the duration of the user’s session.
- The student chooses proceed to registration option and commits the course offering selections. The registration-processing module manages this interaction.
- The application determines whether the user is logged in and if not, either calls the student module to set up a new student or let him/her login.
- The application lets the user revise or cancel the course selections for registration. If the student accepts, the registration processing logic logs the registration, and sends a confirmation message by email.

**Application functional modules:** Below is the list of modules that have been identified and their responsibilities:

- **User registration module:** The application tracks student information. This includes a student id and password and various types of contact information (email addresses, phone number, and so on). Student information is in the database.
- **Course registration module:** The application allows the user to search for courses and be able to display details of individual courses. The catalog includes descriptions of individual course offerings.
- **Assignment module:** The application enables students to upload written assignment to course-teacher. It also allows course-teacher to grade assignment and publish the result to students.
- **Chat module:** A chat module is used for live-time discussions. Chat is a very efficient way to discuss things in live-time.
- **Glossary module:** The application allows a flexible way for students to add to present definitions (and more) that can be linked through the entire class site.
- **Elluminate module:** This is used for synchronous lecture delivery.
- **Lessons module:** This feature allows for the addition of lessons that guide the student based on the student’s answers. It might be helpful to think of a lesson as a kind of flowchart.
- **Wiki:** A wiki is similar to a blog (web log or journal), except everyone can contribute, edit, comment, etc. In general, wiki posts are not approved by the administrator, so the content can be built very quickly.
Workshop module: This creates a workshop space for the class. It is used to facilitate a wide range of peer review options.

Figure 5 shows the interrelationship of the modules corresponding to different tier of the application.

The modular decomposition of the application is reflected in the sub-modules in Figure 5 for the application’s high-level modules shown in Fig. 2.

THE IMPLEMENTATION TECHNIQUE

To keep development simple and under control of its size, the following assumptions are made:

- Application design focuses on the student course registration interaction only. Administration interaction is done manually. Therefore, the EIS tier database tables are pre-populated manually.
- For simplicity, only the courses offered in the coming semester are stored in the Database. However, there is no limit on number of students that can enrol. Theoretically, one or more instructors can offer a course within the same semester, thus, no limit on students.
- There is no restriction on how many courses a student can enrol (a reasonable student is limited by his time and money).
- Instructor is not assigned to departments.
- Once a course is registered, student cannot cancel it. This can only be done by the administrator or instructor through admin interface.

User interface: The user interface consists of a set of screens which is the total content delivered to the browser when the user requests an application URL. In other words, a screen is called whenever the user clicks on one of the application’s URLs. A screen can be composed of several components that constitute different parts of the screen content. From the scenarios and function specifications discussed above, the following interrelated set of screens is developed:

- Entry login screen: This screen is launched and accessed from http://localhost/students/index.html, using a browser. It is a HTML page meant for controlling access to launching the virtual classroom application.
- Course category screen: Here a user can select the appropriate course for self-enrolment after successful logon.
- Registered course screen: From this screen a user has access to all the modules that relate to the course in question such as chat, assignment, forum, etc as already set up by the course administrator or teacher.

Virtual classroom screen: From the Registered Course screen a user can enter the virtual classroom by clicking on ‘Enter Virtual Classroom’ from where he/she can meet with the lecturer online in a prescheduled lecture.

Database schema: The back-end of an application is a collection of state with some rules on how the state changes in response to user interactions. The application system maintains persistent data in MySQL database tables. MySQL is an open-source relational database with capability for middleware development. The database schema, the class diagrams and entity relationships have been omitted in this write-up to conserve space. A number of modules use more than one table to store the needed data in the relational database schema employed by MySQL.

CONCLUSION

In this research, the architecture for an integrated virtual classroom for delivering lectures, issuing and controlling assignments, with chatting, choices and forums is proposed. The proposed architecture takes cognisance of e-learning features such as the White board, Audio and Video features. The Video feature enables one to transmit and receive video broadcasts with others in a classroom session.

Open source facilities were used to develop an application that can give real-time lectures to students in a school system such as University, Polytechnic and College of Education. These open source facilities on asynchronous and synchronous platforms were combined to give a blended real time virtual classroom that mimics the traditional face to face classroom.

RECOMMENDATION

Designing a virtual classroom is an enormous task that requires a lot of resources and time. Tertiary institutions need to wake up and embrace ICT in lecture delivery so that we can be able to train all our youths that are seriously yearning for higher education.

The scope of this study was limited by the recourses at the disposal of the researcher. It would be worthwhile if education agencies collaborate with universities to fund researches on e-learning and virtual classroom instead of spending money in constructing classrooms that will never be adequate for the ever-increasing population.

Recognition given to certificates acquired through distance learning is still very low. Government may have to come up with legislations and guidelines for the implementation of e-learning and virtual classrooms so that the certificates obtained through e-learning can be universally acceptable.
ACKNOWLEDGMENT

The staff of the Computer Centre of Kaduna Polytechnic were instrumental to the setup, implementation and testing of this web based system. Their contribution is hereby acknowledged in this study.

REFERENCES


