Development of an E-Assessment Platform for Nigerian Universities

1O. Akinsanmi, 1Agbaji, 1O.T. Ruth and 2M.B. Soroyewun
1Department of Electrical Engineering, Ahmadu Bello University, Zaria, Nigeria
2Iya Abubakar Computer Centre, Ahmadu Bello University, Zaria, Nigeria

Abstract: This paper presents the development of an e-assessment platform for application in Nigerian universities. Ahmadu Bello University, Zaria, Nigeria was used as a testing ground in 2008. In the age of digitalization that we live in, it has become imperative that a reliable means of assessing students using internet-based technologies be developed. This paper seeks to solve a part of that problem by designing and developing a web application where tests in multiple choice formats will be taken online and graded immediately. The web application relies solely on Microsoft developed technologies. It runs on the Microsoft.net framework, uses the ASP.NET web server, C# as the intermediate language, ADO.NET to interact with the relational database and Microsoft SQL server as the relational database. The work was simulated in the Visual Studio Express environment using Internet Explorer as the web browser. It was then tested on an intranet to ensure connectivity and could be relied upon for a wide application to solve the problem of assessing the teeming population in tertiary institutions.

Key words: ASP.NET, C#, E-assessment, intranet

INTRODUCTION

The 21st century is an age of digitalization where even the simplest of tasks is carried out with the aid of a computer. From shopping to banking; teaching to examinations, talking to video conferencing. The explosion of communication technology via the Internet has brought regular day-to-day tasks to our doorsteps.

With the aim of harnessing this burst of innovative energy and benefiting from all these new technologies provide, an online test engine for the Department of Electrical Engineering would prove to be an invaluable asset to both lecturers and students alike. By storing students data in a database (data is captured on student login) and using cookies and session variables to track a particular user during a session in order not to lose his/her profile and providing all data to lecturers for creation of new, editing and backing up of the data and quick access to students’ results, the whole process of continuous assessments is made a whole lot easier (Green and Mitchell, 2009; Wong, 2009; Dong et al., 2009). A case study of the Department of Electrical Engineering, Ahmadu Bello University, Zaria, Nigeria was carried out in 2008.

Objectives:

- Application seeks to create functionalities for the Examiner to set his/her questions online and for the students to take the test(s) and immediately receive an automated feedback of their test scores.
- This application will allow for the users to see the correction of their tests if they so wish.
- It seeks to provide a forum where students can lay complaints without having to physically see the examiner and for the examiner to attend to the complaints without having crowds outside their doors.
- Can eventually be hosted on the Departmental local intranet server or the University web server.

This work was embarked upon in a bid to bring our universities/colleges up to date with the format of assessments currently being adopted in most standard exams. This application seeks to reduce the strain that is placed on our examiners as in most Nigerian universities, the classes are packed full. Instead of the standard 50 students per class, an examiner may have up to 200 students in his class. Thus multiplying his standard workload by a multiple of 4! This application reduces that strain drastically as the examiner has to worry about setting his questions and retrieving the test scores only since the students tests will be marked and recorded automatically (Green and Mitchell, 2009; Wong, 2009).

The Internet initially began as an experiment in the late 1960’s. The goal was to create a resilient network of computers that would allow people to communicate, share
information and would not collapse on the breakdown of one or more computers (MacDonald, 2006). In the early 1990’s, modems were introduced to open up the Internet to more users, particularly for commercial purposes. In 1993, the first HTML browser was made and from that point onwards, the Internet has literally exploded (Mridula et al., 2002). A Web page consisted of static HTML only, meaning that pages could only be viewed like a brochure and any change to that page would have to be done by hand. HTML 2.0 introduced a new technology called HTML Forms that added some functionality to the static pages including interaction with databases. By introduction of graphical widgets or controls, such as buttons, drop down lists, textboxes etc, Web application developers were able to develop standard input pages that ran on the CGI (Common Gateway Interface) standard, with the ability to store, edit and retrieve data from a database (Mridula et al., 2002). With the CGI standard, the Web Server must launch a completely separate instance of the application each time a Web page was requested for and if the site was a popular one, the Web Server struggled under the load of having many copies of the application running at a time leading to a serious performance dive. To curb this problem, Microsoft developed the ISAPI (Internet Server Application Programming Interface), a higher-level programming model that took care of the performance issues, but introduced great complexities. Microsoft never scrapped the ISAPI, but improved on it to bring a higher-level programming model such as ASP and ASP.NET. ASP is a script-based programming language while ASP.NET is an object oriented programming model that allows a Web application to be put together very easily (MacDonald, 2006). Being a product of Microsoft, ASP.NET is not platform independent and will run only on the Microsoft.NET Framework (MacDonald and Szpuszta, 2005; MacDonald, 2006)

Omuya (2002) designed a website for ABUDEE using HTML. Metibogun (2005) designed and developed a students’ project and research database as a library portal using PHP/MYSQL and Sini (2006) designed and developed a Web portal for ABUDEE Component Equipment Store using PHP/MYSQL. Until now, no attempt has been made to design an online test engine for the Ahmadu Bello University. This work therefore seeks to design and develop a web application where tests in multiple-choice formats will be taken online and graded immediately. The web application relies solely on Microsoft developed technologies.

**MATERIALS AND METHODS**

This study was carried out in the Department of Electrical Engineering, Ahmadu Bello University, Zaria, Nigeria in 2008.

The techniques with which the application was developed are as follows:

- HTML mark-up language
- ASP.NET server side script language
- Microsoft Visual C#
- JavaScript server side script language
- ADO.NET for Database Management
- Microsoft SQL server 2005
- IIS web server

In designing this e-assessment application the following methodologies were employed:

- The intended clients of the web application were first of all determined.
- Study of already existing similar web applications was done to ensure that the web application about to be designed will meet up with standard.
- A survey of the available software packages needed for design and development of the application was carried out in order to determine the best packages to use in terms of scalability, versatility and conversance with the particular technology. The Microsoft.NET package was chosen for its advantage over others in terms of ease of installation, availability of Visual Web developer and SQL Server Express editions that are fully functional subsets of the Visual Studio 2005 and SQL Server 2005 and the fact that the Microsoft.NET provides a world class, powerful set of tools for developing and maintaining data based web applications (Mridula et al., 2002; MacDonald, 2006)
- The Visual Web Developer and SQL server express editions were downloaded and installed.
- Information needed for the design and development of the Registration page, test regulations page, web page for setting the questions and sample test pages in the web application was sourced from the departmental procedures and members of staff (Wong, 2009).
- All information obtained and other supporting functionalities were converted by coding into a format understood by the web browser using ASP.NET scripting, C# language for implementing functionalities along side ASP.NET, HTML for page design and layout, JavaScript for implementing test timing and postback functions, ADO.NET for Querying the database. ADO.NET eases the process of creating, deleting and changing table formats. It allows you to create relationships between tables with the use of Primary and foreign keys, DataSets and DataTable Adapters. ADO.NET also allows for deleting and editing information stored in tables. SQL database acts as the Relational Database that
ADO.NET works with and IIS web server hosts the applications (Belinaso and Hoffman, 2002; Omuya, 2002; MacDonald and Szpuszta, 2005, Anonymous, 2008a, b).

- The application was deployed to a server machine and configured on the server as a virtual directory in IIS so that client computers on the network can access the application via the server.

**RESULTS**

The application’s front-end was developed using basically ASP.NET codes, C# and HTML. ASP.NET and C# are used with MSSQL to output the result of a query to the database and implement the grading functionality (Belinaso and Hoffman, 2002; Davis and Sphar, 2006). In order to query databases and grade test scripts, a database that holds all data including user profiles, test questions and answers, lecturer profiles and other important data was created, then tables for sorting out this data and a connection to this database was made (MSDN, 2008a, b). This application can be broadly classified into two parts; the Student’s section and the Examiner’s section.

**Student’s Section:** The students’ section essentially comprises of the test taking, checking grades and laying complaints functions. The home page contains a link Take the Test that links the user to the registration page (test1.aspx). The student is expected to enter his/her details in the right format. Failure to enter a recognized variable will throw an exception by the required field validator control. Data entered is stored in the test101 table shown in Fig. 1 and is also stored as session variables. This enables the user’s data to be accessible anywhere in the application as long as the browser is not closed.

```
Session["test_course"] = DropDownList1.SelectedValue;
Session["surname"] = TextBox1.Text.ToUpper();
Session["name"] = TextBox2.Text.ToUpper();
Session["reg"] = TextBox3.Text.ToUpper();
Session["passcode"] = TextBox4.Text.ToUpper();
Session.Timeout = 60;
```

Successful registration takes the user to the main test page (Fig. 2) which contains the questions drawn using the sqldatasource from the QuestionT table (shown at design time in Fig. 3), answer options and five radio buttons for selecting an answer. Once the test time has elapsed, the application code retrieves the answers set by the examiner from the test101 table and compares each preset answer with the user’s answer choice. Every

![Fig. 1: Table creation wizard for test101](image1)

![Fig. 2: Test page](image2)
Examiner’s Section: The examiners’ section requires authentication for access. Every examiner’s profile is stored on the ASPNETDB.mdf database. Once access is granted, the examiner can proceed to set his test, view students’ results or complaints. The set the test page implements the following functionalities: it provides textboxes for inputting course code, number of test questions, test pass code, questions, answer options and correct answer (Fig. 5). All these are saved in the QuestionT table through the use of sqldatasource control. A counter shows the examiner how many questions he has set so that he can keep track of his questions. The questions are saved in the QuestionT table. The view results page allows the examiner to view the entire student’s result and export it to Microsoft Excel and edit if necessary (Fig. 6). There is also provision for editing already set question through the edit questions link. So that in the event that the Examiner changes his mind about a question, he can edit it before the test is taken. The View Complaints link, is the channel through which the examiner can view complaints made online by the students and attend to them accordingly. The Create new lecturer profile link is located in the Examiner pages for security reasons. In order not to allow unscrupulous elements registering as Examiners, the link is accessible to only registered users and only a registered user can register another user thereby providing the application with a means of verifying its registered users (Fig. 7).

CONCLUSION

This application has demonstrated a possible solution to the problem of carrying out course assessments for a large number of students per class. Such large classes can...
be found majorly in Africa and this is an aspect of the rural nature of our Nigerian educational system. The ideal solution to urbanizing our educational systems in all areas would be to build more schools with subsidized fees so it can be affordable by all. But that solution is far fetched and it would easier and still very relevant- if the schools are eventually built- to install e-assessment packages. If e-assessment and e-learning web applications are run in
our schools and colleges nationwide, it would greatly reduce stress on both students and examiners alike, keep us up to date with the latest technology used in educational sectors and also provide an opportunity for Nigerian students to be more conversant with fundamental concepts of ICT.

**REFERENCES**


Metibogun, S., 2005. The design and development of ABUDEE students project and research database library using PHP/MYSQL” Final Year Project (B. Eng.) Electrical Engineering ABUDEE Zaria.


