

Design and Implementation of Secure Browser for Computer-Based Tests

¹Onyemaobi Bethram Chibuzo and ²Dauda Olorunkemi Isiaka

^{1,2}Department of Computer Science

College of Information and Communication Technology,
Salem University, Lokoja Kogi Nigeria.

Abstract:- This work is concerned with the development of a browser for computer based examination platforms. It focused on trends and mostly used browsers for online computer based examination and carried out a critical review of the browsers used for current computer based test system employed in Nigeria Universities. An alternative application to provide solutions to the current challenges identified in the existing system was then developed. The new application was designed using the Extreme programming methodology and implemented using Microsoft Visual Basic on a Windows 7 system. We created user-friendly interfaces driven by the Visual Basic codes. The application has amongst other features, disabled access to control box of the browser window and the Alt + F4 button as shortcut in closing browsers.

Keyword:- computer-based test; CBT; browser; application; data flow diagram; DFD; wireframe.

I. INTRODUCTION

Nigeria is a developing nation and as such, has flares for embracing latest technologies and trends. However, the developed nations are taking preventive measures against the odds of these technologies and trends. The creation and diffusion of knowledge has become increasingly important factors in competitiveness and comfortable living. So, the effective evaluation and measurement of knowledge and skills must be done through good processes and better tools, so as to derive objective and correct results that truly represent the actual strength and ability of students or prospective applicants. These correct and objective results would in turn inform better decision for the institution as to who is good or better for admission or grade points determination in the case of university students.

(Tuckman, 1975), defined performance as the apparent demonstration of understanding, concepts, skills, ideas and knowledge of a person, and proposed that grades clearly depicts the performance of a student. Hence their academic performance must be managed efficiently, keeping in view all the factors that can positively or negatively affect their educational performance and thus, form the basis of the institutions' decisions. Use of technology such as Internet, network of computer systems and computer-based test (CBT) applications loaded through

tools like browsers are most important factors that can be used to evaluate students to determine their actual performance.

The rapid adoption is due to the numerous advantages the institutions derive from CBT over the traditional paper and pen testing considering the large population of students. Some of the advantages include: increased delivery, administration and scoring efficiency, improved test security, consistency and reliability, faster response rate to mention a few (Riku, *et al.*, 2001; Wilks, 2002). This development calls for periodic evaluation of factors that determine students' acceptability of this new mode of assessment in order to ensure its fairness to students. As primary users of CBT, students' opinions are also needed on CBT systems to enable examiners and developers know areas requiring improvement from the users' perspective.

Some students have become very smart because of their exposure and good knowledge of these technologies and trends, such that they would engage in some activities during the computer-based testing (CBT) that are inimical to the process of the evaluation and the output does not actually represents the performance, skills and ability of these students. These activities ranges from a student minimizing the browser of his or her test page to use the search engines to get answers for some questions, if the system is Internet-enabled, or insert devices that can create Internet network for the students to browse for answer, or even taking screenshot and sending the questions to a partner somewhere for assistance and answers revert back to him or her.

This paper developed a browser that can be used for computer-based tests organized and managed by Nigerian Universities as exemplified in Federal University Lokoja (FUL). It is a system-based tool that is able to load and allow students take their Computer-Based Test (CBT) and not have access to the desktop and other tools outside the browser by virtue of certain challenges in the selected scope of Study. It has been observed that some factors like distractions, and improper invigilation have deprived the effective and actual performance, knowledge and understanding of applicants for admission during the Post-Unified Tertiary Matriculation Examination (PUTME) and the undergraduate students taking Continuous Assessment (CA) through the Computer-Based Tests. Most Computer-Based Tests or Exams contain recycled questions that have

been made available in various websites – leading to a high level of malpractices.

The paper propose to achieve the following:

- Create a module for the administrator to be able to manage and exit the browser.
- Reduce the level of examination malpractices and improve the credibility of the examinations via a module to disable control box in order to disallow users/students access other tools on the computer systems during examinations and tests.
- Provide a friendly, easy to install and use browser for CBT/examination platforms for the Universities.

II. RELATED WORKS

➤ *The Concept of Web Browser*

A web browser is software program that allows users to locate, access and display web pages. A variety of web browsers are available with different features, and are designed to run on different operating systems, for example, Internet Explorer from Microsoft, Firefox from Mozilla, Chrome from Google, Safari from Apple Inc., Opera, Safe Exam Browser (SEB) etc. The Safe Exam Browser according to Schneider *et al.* (2010) is a Web Browser environment to carry out online-exams safely. The software changes any computer into a secure workstation. It regulates the access to any utilities and prevents students from using unauthorized resources. However, installation of SEB requires an amazing memory of passwords and thus makes users’ experience not too good during installation.

The first graphical Web browser, Mosaic (<http://www.boutell.com/newfaq/history/fbrowser.html>), was released in 1993, making the World Wide Web accessible to everyone, and helping to launch an information explosion that continues to this day. About a

year later, Marc Andreessen founded Netscape, which released Navigator as its flagship product. In the following year, Microsoft joined the race by releasing its Web browser; Internet Explorer. Microsoft’s Internet Explorer (IE) is the most widely used web browser, and the IE’s global usage is reported as 93.9% share in May 2004 according to OneStat.com in Park *et al.* (2005).

According to Nielson *et al.* (2011), Modern Web browsers are composed of several parts. Each browser must have a rendering engine to create the layout and appearance of a Web page, a scripting engine to interpret and execute JavaScript (or similar) scripting code on a Web page, and a user interface that includes page navigation controls, as well as many other features (e.g., history, preferences, plugins) created by the browser designer. When a typical user selects a preferred browser, they most likely base their decision on the interface, since it is the most visible distinguishing feature. However, the other aspects, which are more technical in nature, should not be ignored. Both the rendering engine and scripting engine could be graded on multiple aspects. In this paper, we focus mainly on performance, correctness and security.

Accessing a web page on the server or Internet by the Hypertext Transfer Protocol (HTTP), there is communication between two computers. On one computer there is a software program known as a browser, on the other is a software program known as a web server. The browser sends a request to the server and the server sends a response to the browser. The request contains the name of the page that is being requested and information about the browser that is making the request. The response contains the page that was requested (if it is available), information about the page and information about the server sending the page; see Figure 1

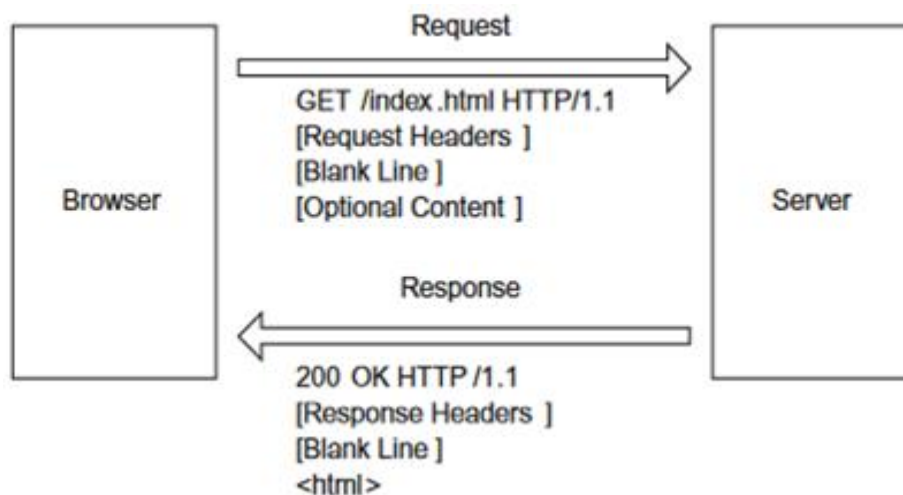


Fig 1:- The request and response have specific formats, as specified by the HTTP protocol. Source: https://doi.org/10.1007/978-1-84628-863-0_1

YEAR	WEB BROWSERS
1991	WorldWideWeb (Nexus)
1992	ViolaWWW, Erwise, MidasWWW, MacWWW (<i>Samba</i>)
1993	Mosaic, Cello, Lynx 2.0, Arena, A Mosaic 1.0
1994	IBM WebExplorer, Netscape Navigator, SlipKnot 1.0, MacWeb, IBrowse, Agora (<i>Argo</i>), Minuet
1995	Internet Explorer 1, Internet Explorer 2, Netscape Navigator 2.0, OmniWeb, UdiWWW, Grail
1996	Arachne 1.0, Internet Explorer 3.0, Netscape Navigator 3.0, Opera 2.0, PowerBrowser 1.5, Cyberdog, Amaya 0.9, AWeb, Voyager
1997	Internet Explorer 4.0, Netscape Navigator 4.0, Netscape Communicator 4.0, Opera 3.0, Amaya 1.0
1998	iCab, Mozilla
1999	Amaya 2.0, Mozilla M3, Internet Explorer 5.0
2000	Konqueror, Netscape 6, Opera 4, Opera 5, K-Meleon 0.2, Amaya 3.0, Amaya 4.0
2001	Internet Explorer 6, Galeon 1.0, Opera 6, Amaya 5.0
2002	Netscape 7, Mozilla 1.0, Phoenix 0.1, Links 2.0, Amaya 6.0, Amaya 7.0
2003	Opera 7, Apple Safari 1.0, Epiphany 1.0, Amaya 8.0
2004	Firefox 1.0, Netscape Browser, OmniWeb 5.0
2005	Opera 8, Apple Safari 2.0, Netscape Browser 8.0, Epiphany 1.8, Amaya 9.0, AOL Explorer 1.0, Maxthon 1.0, Shiira 1.0
2006	Mozilla Firefox 2.0, Internet Explorer 7, Opera 9, SeaMonkey 1.0, K-Meleon 1.0, Galeon 2.0, Camino 1.0, Avant 11, iCab 3
2007	Apple Safari 3.0, Maxthon 2.0, Netscape Navigator 9, NetSurf 1.0, Flock 1.0, Conkeror
2008	Google Chrome 1, Mozilla Firefox 3, Opera 9.5, Apple Safari 3.1, Konqueror 4, Amaya 10.0, Flock 2, Amaya 11.0
2009	Google Chrome 2–3, Mozilla Firefox 3.5, Internet Explorer 8, Opera 10, Apple Safari 4, SeaMonkey 2, Camino 2, surf, Pale Moon 3.0
2010	Google Chrome 4–8, Mozilla Firefox 3.6, Opera 10.50, Opera 11, Apple Safari 5, K-Meleon 1.5.4, xxxterm
2011	Google Chrome 9–16, Mozilla Firefox 4-9, Internet Explorer 9, Opera 11.50, Apple Safari 5.1, Maxthon 3.0, SeaMonkey 2.1–2.6
2012	Google Chrome 17–23, Mozilla Firefox 10–17, Internet Explorer 10, Opera 12, Apple Safari 6, Maxthon 4.0, SeaMonkey 2.7-2.14
2013	Google Chrome 24–31, Mozilla Firefox 18–26, Internet Explorer 11, Opera 15–18, Pale Moon 15.4-24.2.2, Apple Safari 7, SeaMonkey 2.15-2.23
2014	Google Chrome 32–39, Mozilla Firefox 27–34, Opera 19–26, Pale Moon 24.3.0-25.1.0, Apple Safari 8, SeaMonkey 2.24-2.30
2015	Google Chrome 40–47, Microsoft Edge, Mozilla Firefox 35–43, Opera 27–34, Pale Moon 25.2.0-25.8.1, Vivaldi
2016	Google Chrome 48–55, Mozilla Firefox 44–50, Microsoft Edge 14, Opera 35–42, Pale Moon 26.0.0-27.0.3, Apple Safari 10, SeaMonkey 2.24–2.30
2017	Basilisk, Google Chrome 56–60, Microsoft Edge 15, Mozilla Firefox 51–55.0.2, Opera 43–45, Opera Neon, Pale Moon 27.1.0-27.6.2
2018	Pale Moon 27.7.0-28.0.0

Table 1:- Computer web browsers by year of release of major version.

➤ Related Research on Knowledge Management

Web-based Examination System is an effective solution for mass education evaluation (Zhenming *et al.*, 2003). They developed a novel online examination system based on a Browser/Server framework which carries out the examination and auto-grading for objective questions and operating questions, such as programming, operating Microsoft Windows, editing Microsoft Word, Excel and PowerPoint, etc. It has been successfully applied to the distance evaluation of basic operating skills of computer science, such as the course of computer skills in Universities and the nationwide examination for the high school graduates in Zhejiang Province, China.

Ayo *et al.* (2007) proposed a model for e-Examination in Nigeria where all applicants are subjected to online entrance examination as a way of curbing the irregularities as proposed by the Joint Admissions Matriculation Board

(JAMB), the body saddled with the responsibility of conducting entrance examinations into all the Nigerian universities. This model was designed and tested in Covenant University, one of the private universities in Nigeria. Their findings revealed that the system has the potentials to eliminate some of the problems that are associated with the traditional methods of examination such as impersonation and other forms of examination malpractices. Another 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 (Akinsanmi *et al.*, 2010). The author further explained that, 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, but to be loaded by any of the famous

browsers like Mozilla Firefox, Google Chrome, Safari, Internet Explorer, etc. Based on the development of e-learning in the National Open University in Nigeria (Ipaye, 2009), discussed the process of establishing e-learning environment.

➤ *The Computer-Based Testing*

According to Noyes *et al.* (2008) Computer-Based Testing is the act of using Information Technology in various forms of assessment such as educational assessment, health assessment, psychological assessment, etc. This way of testing is important because it can measure different skills or sets of knowledge in order to provide new and better information about one's abilities. The tool that assists the user to display and take the examination is the Web Browser. However, if the Web Browser is not secure and developed to guard against sharp practices, the effectiveness of the outcome of the testing would mislead and definitely inform wrong decision during selection of best performed students either for the purposes of admission or scholarship, as the case may be. In other words, Computer-Based test is simply taking a test on a computer, using the keyboard and mouse to "flip through" and "answer" your test questions that appear on a computer monitor.

An online website for tutoring and e-examination of economic course aimed to present a novel software tool can be used for online examination and tutorial application of the syllabus of economic course (El Emery and Al Sondos, 2006). Also, among the main interests of the paper is to produce a software through it we make sure that students have studied all the concepts of economics. So, the proposed software is structured from two major modules: The first one was an online website to review and make self-test for all the material of economic course. The second part is an online examination using a large database bank of questions through it the level of students can be evaluated immediately and some statistical evaluations can be obtained. The developed software offers the following features:

- Instructors could add any further questions to maximize the size of the bank of questions.
- Different examinations for each student with randomly selected questions from the bank of questions can be done.
- Different reports for the instructors, students, classes...etc. can be obtained.
- Several students can take their exams simultaneously without any problem inside and outside their campus. The proposed software has been designed to work base on the client/server architecture.

Electronic exam is a difficult part of e-learning security (Huszti and Petho, 2008). The paper describes a cryptographic scheme that possesses security requirements, such that authenticity, anonymity, secrecy, robustness, correctness without the existence of a Trusted Third Party. The proposed protocol also provides students a receipt, a proof of a successful submission, and it is based on existence of anonymous return channels. Another research

work proposed a theoretical approach that incorporates available fingerprint biometrics authentication technologies in conjunction with e-learning environments to curb unethical conduct during e-learning exam taking (Levy and Ramim, 2007). The proposed approach suggests practical solution that can incorporate a random fingerprint biometrics user authentication during exam taking in e-learning courses. Doing so is hypothesized to curb exam cheating in e-learning environments

III. ANALYSIS

The existing system is the usage of open source web browsers to load Computer-Based tests, such as Firefox from Mozilla, Internet Explorer from Microsoft, Chrome from Google, Safari from Apple, Opera, etc. These existing systems/tools do not give the administrators or managers of Computer-Based Tests the room to configure and customize the web browsers to suite test purposes. You can neither disable nor enable some tools peculiar to Computer-Based testing, like disabling control buttons, so that the applicants or undergraduate students do not minimize or maximize the browser window at will. Also, surfing the net to get answers while taking Computer-Based tests cannot be totally controlled. It will be tedious to maintain strict supervision with these limitations and the human effort is kind of more here. This system requires extra efforts in preserving the sanity of the tests and evaluation. Suppose the applicants or undergraduate students were able to surf the Internet to get answers for their various questions during the test, the aim of the evaluation has been defeated. So, it will be difficult to ascertain the best students in terms of performance, because the result of the testing is not objective and the university might end up admitting weaker students, and also the undergraduate students' results would not represent the true level of competence of the students. The existing system is strictly to install and use. The administrators do not have control over the limitations stated above and as such, most Computer-Based Tests loaded with these open source web browsers are prone to cheating and exam malpractice.

The proposed system contains the findings of the existing system and recommendation to overcome the problem and limitations of the existing system. The new system will reduce cheating, exam malpractice, reduce the stress of supervision during the Computer-Based Tests and help to provide near accurate output in terms of the measure of knowledge, skills, ability and competency.

The functional requirements of this system include;

- The system will be able to load all Computer-Based Test software so as to carry out exams safely.
- The system will allow controlled use of additional incorporated application like Calculator for arithmetic purposes.
- The system will prevent students from surfing the internet during Computer-Based Tests.
- The system will prevent applicants or undergraduate students taking tests from minimizing, maximizing and closing the test pages.

- The system will only allow the administrators with required privileges to manage it.

Extreme programming, a type of agile development methodology is used in the design of the system. The agile development focuses on streamlining the software development life cycle (SDLC). Also, much of the modeling and documentation overhead is eliminated, instead face-to-face communication is preferred. Agile methodology is also useful in the development of systems

with short time schedule and also system with unclear user requirements. There are several popular approaches to agile methodology, but for the implementation of this System, the Extreme Programming (XP) approach was used as shown in figure 2. Extreme Programming begins with the user describing what the system needs to do. Then the programmer codes in small modules and test to meet the users' need. Users are usually expected to be available to clear up questions.

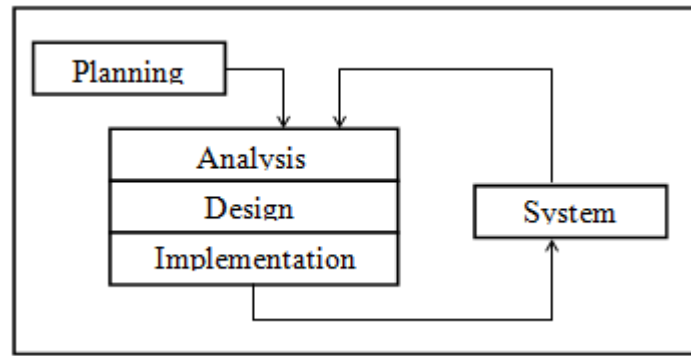


Fig 2:- Diagram for Extreme Programming methodology

IV. SYSTEM DESIGN AND IMPLEMENTATION

The system is designed to use the Client architecture. This means that, the system would be deployed and run on all the clients within the network to load CBT platforms, so as to allow the users/students take the CBT/Exams safely.

- *Conceptual design of the browser application software*

The conceptual design of the browser application software was carried out using the activity diagram. Activity diagrams are graphical representations of workflows depicting stepwise activities and actions with support for choice, iteration and concurrency. In the unified modeling language, activity diagrams can be used to describe the business and operational step by step workflows of components in a system. An activity diagram shows the overall flow of control. Figure 3 presents the activity diagram for the browser application system developed.

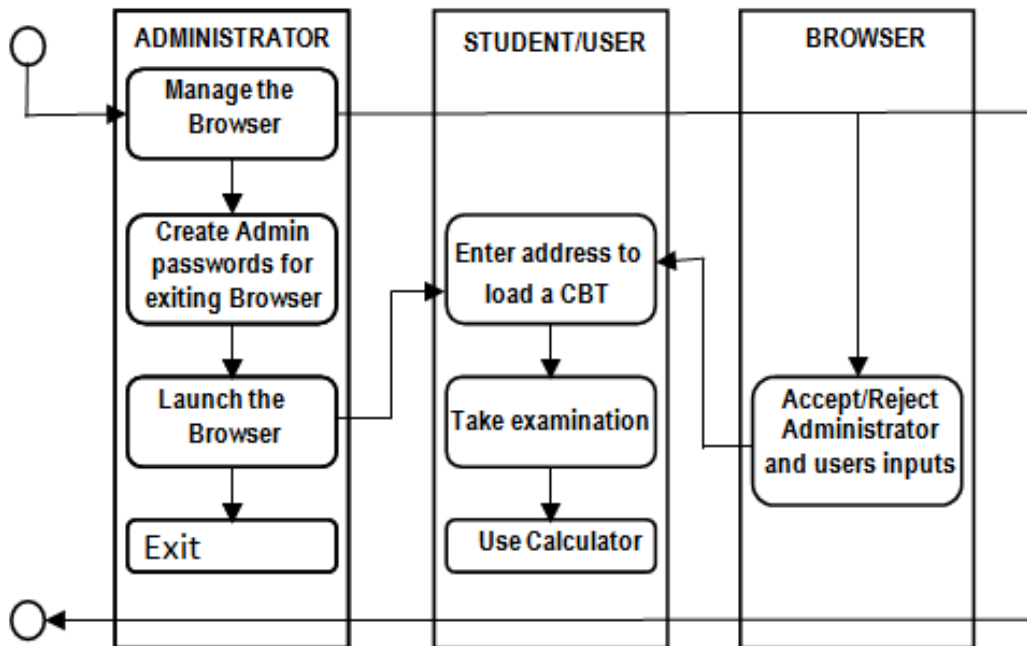


Fig 3:- Activity diagram for the Browser application model

➤ *Data Flow Diagram (DFD) for the browser application system*

A data flow diagram (DFD) uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among the functions. Starting with a set of high-level functions that a system performs, a DFD model hierarchy represents various sub-functions. The data flow diagram depicted in figure 4 below shows the relationship among the entities in the Browser application system. The entity “STUDENT/USER” can load the CBT application and take examination after he or she gains access to launch the browser system. The entity “ADMINISTRATOR” is saddled with the responsibility of managing the browser application, launch the browser, exiting/closing the browser and setting the default password for exiting of the browser system. The entity “BROWSER APPLICATION” is responsible for authenticating the users of the browser system and also provides the functionality for the student to load and run CBT examination platforms.

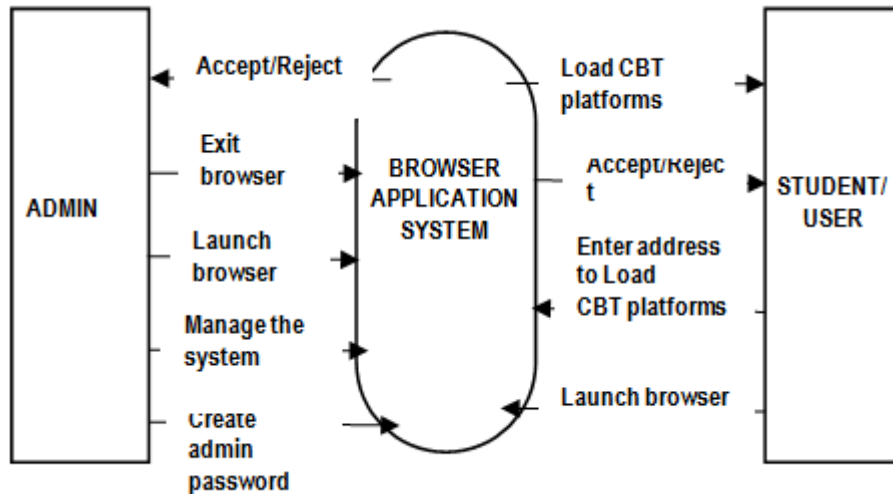


Fig 4:- Data-Flow-Diagram (DFD) of the Browser application model

➤ *Use case diagrams for the browser application system*

Use case diagram for each entity present in the Browser application system is presented here. These include use case diagrams for the system administrator and student/user. The use case diagram for the administrator is presented in figure 5. It shows the activities that are required of the administrator including the managing of the system, launching the browser, exiting the browser and creation of admin passwords for exiting the browser.

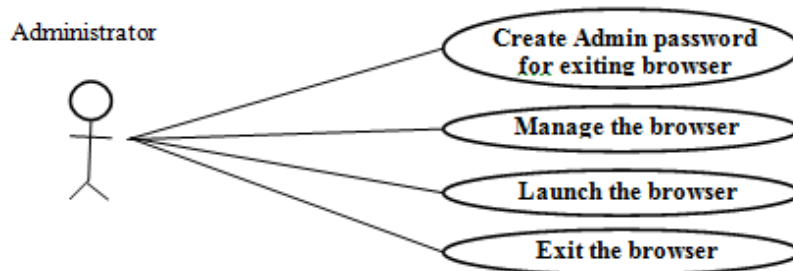


Fig 5:- Use case diagram for the administrator

The use case diagram for student shows that the student can access the system and write his or her examination. Figure 6 presents the use case diagram for the student.

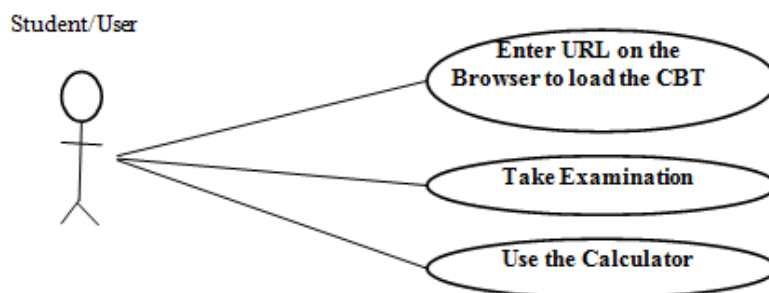


Fig 6:- Use case diagram for the students/users

➤ *Wireframe of the system interfaces design*

The interface is the medium/point at which users interact and input data into the system/browser and the browser in turn presents response back to the user as requested. The browser would have four interfaces for different purposes as shown in figure 7. They are namely;

- Students/Users’ interface (Form 1): It is also called the homepage/start page. This is the page for the students/users to load the CBT platform by entering the URL of the CBT platform on the address bar of the page and take examination/Continuous Assessment.
- Administrator’s Login interface (Form 2): This is an authentication interface that demands the administrator to enter his or her admin username and passwords to be able to access the admin panel.
- Administrator’s interface (Form 3): This is also called the admin panel. If the administrator is able to login successfully, he or she would be able to manage the browser and also become the only person to exit the browser when necessary.
- Calculator’s interface (Form 4): This is for arithmetic calculations that may arise in the course of the examination or continuous assessment test. It is within both the user’s panel and the administrator’s panel. Most of the basic operations for arithmetic were captured.

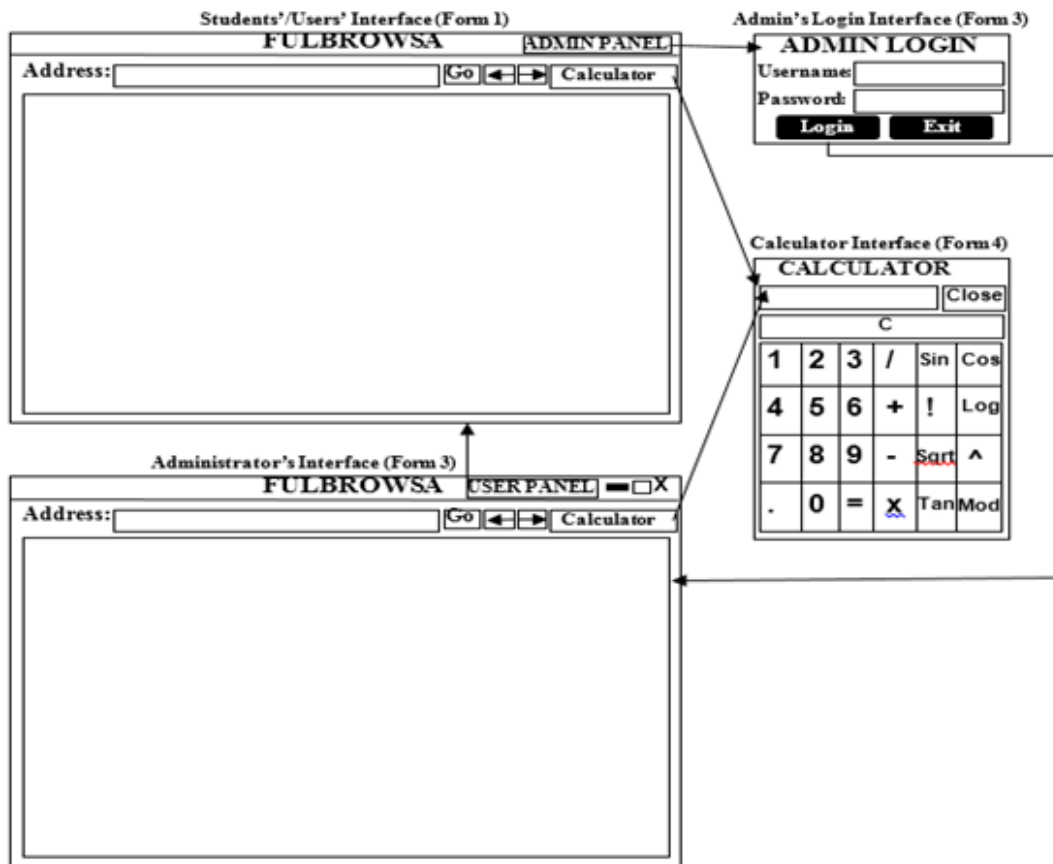


Fig 7:- Wireframe of the system interfaces design

V. RESULTS AND DISCUSSIONS

Visual Basic Programming Language of Microsoft Visual Studio 2010 suite is the IDE used in implementing the client code and interface of this application. Visual Basic editor is a fast and powerful in creating and executing applications for Windows, combining features of a fully-packed Windows form properties. Microsoft Visual Studio 2010 suite is one of the most complete all-in-one software for coding in Visual Basic, C++, C# and C development languages with tools for debugging (Onyemaobi and Ajah, 2017), validating, reusing, navigating and formatting your code. With Visual Basic one can code smarter, save time and increase productivity.

The program source code is published, deployed to run on any Windows Operating System ranging from Windows 2000/XP/VISTA/7/8/8.1/10.

The unit testing approach was adopted in testing the written codes. The procedure adopted for the unit test is;

- The module interface is tested to ensure that information properly flows into and out of the program unit under test.
- The local data structure is examined to ensure that data declared and stored within the source code maintained its integrity during all steps in an algorithm’s execution.
- All the statements are executed at least once and error handling paths are tested

The screenshots of the developed browser interfaces are shown below:

- Students/Users’ interface (Form 1): It is also called the homepage/start page. This is the page for the students/users to load the CBT platform by entering the URL of the CBT platform on the address bar of the page and take examination/Continuous Assessment (see figure 8).

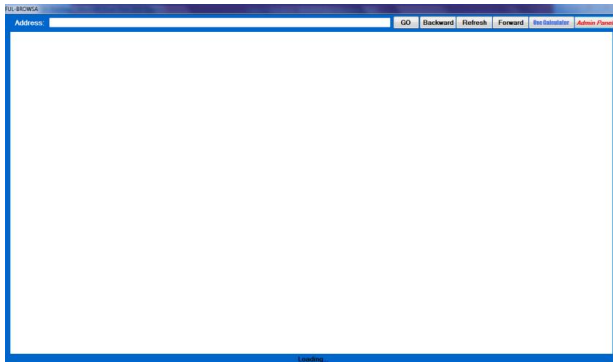


Fig 8:- Screenshot of the students'/users' interface of the developed browser

- Administrator’s Login interface (Form 2): This is an authentication interface that demands the administrator to enter his or her admin username and passwords to be able to access the admin panel as depicted in figure 9.

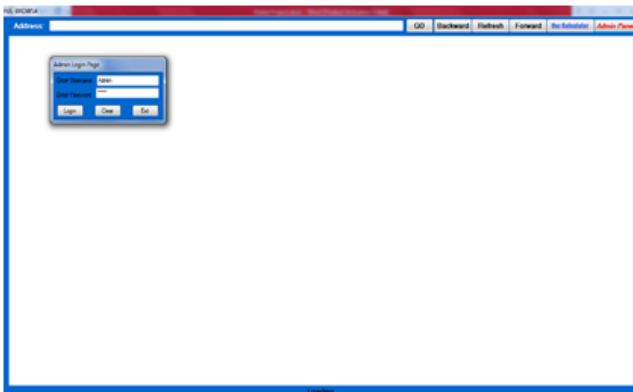


Fig 9:- Screenshot of the administrator's login interface of the developed browser

- Administrator’s interface (Form 3): This is also called the admin panel as shown in figure 10. If the administrator is able to login successfully, he or she would be able to manage the browser and also become the only person to exit the browser when necessary.

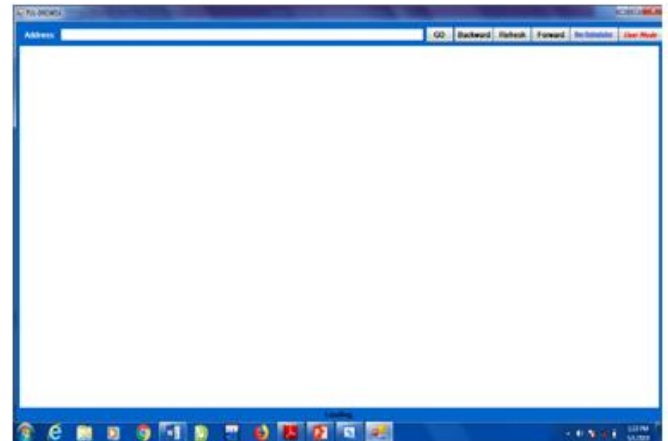


Fig 10:- Screenshot of the administrator's interface of the developed browser

- Calculator’s interface (Form 4): This is for arithmetic calculations that may arise in the course of the examination or continuous assessment test. It is within both the user’s panel and the administrator’s panel. Most of the basic operations for arithmetic were captured as seen in figure 11.

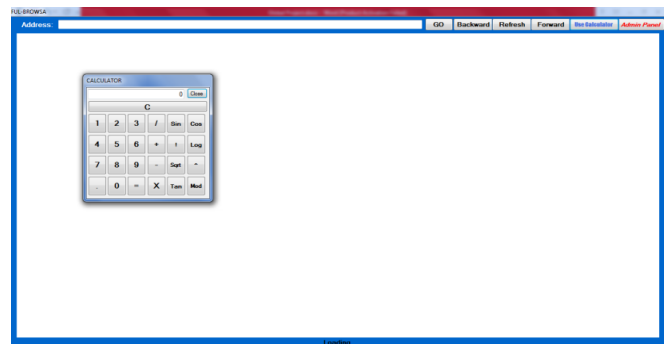


Fig 11:- Screenshot of the calculator's interface of the developed browser

The developed application needed the following hardware for effective operation:

- Pentium IV and above system.
- The Random-Access Memory (RAM) should be at least 1GB
- Enhanced Keyboard.
- At least 200GB Hard Disk.
- E.G.A/V.G.A. coloured Monitor.
- An uninterruptible Power Supply (UPS) Units.
- Voltage Stabilizer: This facilitates the regulation of voltage needed by the computer system in order to avoid electrical damage of the system.

The software requirement for this system is:

- Windows XP/Vista/Window 7/8/10 Operating System.

VI. CONCLUSIONS

In this paper, a browser application for CBT system is developed for Nigeria Universities. The browser is application software for loading CBT examination system that delivers questions set by the Instructor to the student; generates and create the report of the results of students who take the examination. Challenges like examination malpractices, delay or postponements of assessments due to maintenance processes, inadequate supervision by invigilators, and sharp practices of surfing the net while taking examinations by students will automatically be eliminated following the adoption of this browser.

The system is designed using the Agile Methodology through the Extreme Programming approach and Unified Modeling Language was used to bring the view to real life situation. Top-to-down approach was adopted as the implementation approach for the development. This involves breaking complex system into subsystems and then into modules for easy study and understanding. The system architecture is basically divided into three basic parts. The first is the front end that shows the user interface designed with Visual Basic Windows Form, at the middleware are the Visual Basic codes; which provides the connectivity between the front end and the back end. The user interfaces are interactive and provisions are made for security of administrators data stored within the codes. The installation and use of the browser is relatively simple and the technical knowledge requirement for its usage is relatively minimal. The system is generic, meaning that it can be further expanded or more features can be added into the browser to strengthen and make it better. The software is meant to be used for all kinds of Computer-Based Tests conducted or managed by Universities in Nigeria. However, it is limited to Post-UTME Examinations for prospective applicants for admission and Continuous Assessment Tests for admitted students on courses with large population of students.

Having developed the browser for computer based examination platforms/systems, it can be subjected to modification of its features as time goes on. This is a very creative web-browser application that can load various CBT platforms/systems. It is platform independent application software that can also be evoked when necessary reference is needed. As a result of the administrator's authentication feature of the system, the system cannot be closed by anyone without the privileges, and this guarantees that the students would have access only to the system for the purpose of taking an assessment.

RECOMMENDATIONS

We wish to recommend the following:

- Review the scope of the application to ensure improvement in the developed system.
- Adoption and implementation of the application to improve the credibility of CBT assessments in the Universities.

- Apart from the universities other CBT Centres can adopt and modify the browser since it was found to be cost effective.
- Implementing this tool or application will bring about an objective and near accurate result of the performance, knowledge, skills and ability of students that would undertake the Computer-Based Tests in terms of Post-Unified Tertiary Matriculation Examination (PUTME) and the undergraduate students Continuous Assessment (CA).
- Implementing the application makes Computer-Based Test system free from what would aid or allow sharp practices. It is significant because it simplifies the supervision and delivery of good measure of knowledge, ability and skills, which would in turn allow timely and effective decision making.
- The whole process of evaluation through Computer-Based Test system, in which the students or applicants are focused on the test page only without any reason or way to indulge in sharp practices would help both staff and student work smarter, minimize malpractice and distractions, and also produce students that meet standard professional requirements.

REFERENCES

- [1]. Akinsanmi, O. A., Olatunji, T. R., and Soroyewun, M. B. (2010). Development of an E-Assessment Platform for Nigerian Universities, *Research Journal Applied Sciences, Engineering and Technology*; 2(2): 170-175, ISSN: 2040-7467.
- [2]. Ayo, C. K., Akinyemi, I. O., Adebisi, A. A., and Ekong, U. O. (2007). The Prospects of E-Examination Implementation in Nigeria, Department of Computer and Information Sciences, Covenant University, Ota, NIGERIA. *Turkish Online Journal of Distance Education-TOJDE*, Vol 8, No. 4(10), 125-135, ISSN 1302-6488.
- [3]. Bongju, L., and Kim, L. (2007). *Browser-Server Communication. In: Web Development with Java*. Springer, London. https://doi.org/10.1007/978-1-84628-863-0_1
- [4]. Boutell.com, "WWW FAQs: What was the first Web browser?". <http://www.boutell.com/newfaq/history/fbrowser.html>
- [5]. Huszti, A., and Petho, A. (2008). "A Secure Electronic Exam System", *Informatika a felsőoktatásban*. 1-7.
- [6]. Ipaye, B. (2009). "E-Learning in a Nigerian Open University, National Open University of Nigeria. 1-11.
- [7]. Levy, Y., and Ramim, M. M.(2007). *A Theoretical Approach for Biometrics Authentication of E-Exams*. Nova Southeastern University, USA. 93-101.
- [8]. Nielson, J., Williamson, C. and Arlitt., M. (2011). Benchmarking Modern Web Browsers: Journal of Computer Scientists of Calgary.
- [9]. Onyemaobi, C. B., and Ajah, I. A. (2017). 'Comparative analysis of web development languages performances', *Int. J. Web Science*, Vol. 3, No. 1, pp.16-31.

- [10]. Park B., Hong S., Oh J., and Lee H. (2005) Defending a Web Browser Against Spying with Browser Helper Objects. In: Kantor P. et al. (2005), (eds) Intelligence and Security Informatics. Lecture Notes in Computer Science, vol 3495. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11427995_85
- [11]. Wilks, S., and Ricketts, C. (2002). What factors affect student opinions of computer-based assessment? In Myles, D. (Eds.), CAA 2002 International Conference, University of Loughborough, Retrieved from <http://caaconference.com> on 2nd of June, 2012.
- [12]. Riku, S., Lauri, M., and Ari, K. (2001). Fully automatic assessment of programming exercises. ACM Digital Library.
- [13]. Schneider, S. and Meys, F. (2010). *Development of Safe Exam Browser*. ETH Zurich, Educational Development and Technology (LET): University of Giessen.
- [14]. Trautschold M., and Mazo G. (2009). *Understanding the Web Browser*. In: *Pro JavaScript™ RIA Techniques*. Apress. https://doi.org/10.1007/978-1-4302-1935-4_3
- [15]. Tuckman, H. (1975). Teacher Effectiveness and Student Performance. *J. Econ. Educ.*, 34-39.
- [16]. Wikipedia, “List of Web Browsers”, http://en.wikipedia.org/wiki/List_of_Web_browser.
- [17]. Wikipedia, “Web Browser”, http://en.wikipedia.org/wiki/Web_browser.
- [18]. Noyes, J. M., and Garland, K. J. (2008). Computer-vs. paper-based tasks: Are they equivalent?. *Ergonomics*, 51(9), 1352-1375.
- [19]. Zhenming, Y., Liang, Z., and Guohua, Z. (2003), “A Novel Web-Based Online Examination System for Computer Science Education”, 33rd ASEE/IEEE Frontiers in Education Conference, S3F-7-S3F-10.