

A PERFORMANCE ANALYSIS FOR EVALUATION OF PROGRAMMING LANGUAGES BASED ON MOBILE COMPUTING FOR NIGERIA

¹Aishat Mohammed (aishamohammed28@gmail.com), ²Souley Boukari (bsouley2001@yahoo.com), ¹Samson Henry Dogo (doqojankasa@gmail.com) & ¹Safiyanu Ahmad (ahmadsafiyanu100@gmail.com)

¹Department of Mathematical Sciences, Kaduna State University, Kaduna- Nigeria

²Department of Mathematical Sciences, Abubakar Tafawa Balewa University, Bauchi-Nigeria

Corresponding Author's E-mail: aishamohammed28@gmail.com

ABSTRACT

In Nigeria, several programming Languages exist from general purpose to special purpose programming languages that are used in one application domain. People always find difficulties about which programming language should be learnt and adopt to develop particular software. In this paper, three (3) most commonly used programming languages C#, Vb.net and Java were compared both theoretically and empirically based on data security, data connectivity and data Transfer. The algorithms for each aforementioned criterion were implemented in a mobile device, specifically, Android and Window phones. To know the actual performance of these three chosen programming languages, a simulation study was conducted by testing different size of image files on each programming languages based on the set criteria. The results show that of the three programming languages considered, C# is suitable for data security. Vb.net is appropriate for the data connectivity such as download scheme. Finally, Vb.net is suitable for data Transfer using upload scheme.

Keywords: Mobile device, Data Security, Connectivity, Transfer, Speed, Programs

1. INTRODUCTION

The development in the technology of programming languages is rapid. Simply, programming as a problem has only arisen since computer machines were first created. The magnitude of the problem is however relative to the size (and complexity) of the computer machine used (Cook, 1999). To program a computer system, a programming language is required. A programming language should be reasonably natural for solving problems, at least problems within its intended application area. For example, a programming language whose only data types are numbers and arrays might be natural for solving numerical problems, but would be less natural for solving problems in commerce or artificial intelligence. Conversely, a programming language whose only data types are strings and lists would be an unnatural choice for solving numerical problems (Chen, 2010). In Mobile Computing, Applications Software relies on the operating systems for most I/O tasks. Different operating systems uses different methods to access hardware, making application coded in languages like C++ unable to run under other operating systems than what they were written for. One way to solve this problem is to introduce a software layer between the application and the operating system. Mobile computing is a new computing paradigm in which mobile devices are used to access information from anywhere, at any time. Wireless network greatly enhances the utility of a portable computing device. The field of mobile computing is merger of the

portable computing device and wireless communication with the aim of providing seamless computing environment for mobile users.

So many works have been done with the aimed to achieve the best or most suitable programs to solve particular task as a result of different programming languages that exist leading to a difficulties to people to make strategic choice but it was always revealed that whatever approaches or methods Authors adopted to solve the problems in comparing programming languages for a certain task, there is always weaknesses such as quality assurance software, poor development practice, maintainability of software development high, dependability and Safety of the software was low. To overcome these problems we considered this research in Mobile Computing, since Mobile computing is a new computing paradigm in which mobile devices are used to access information from anywhere, at any time. In this paper however, having stated that mobile computing is a good technique, we measured and compared three of its major drawbacks or challenges: Data Security, Data Connectivity and Data Transfer as an extension to the work of (slaiman *et al.*, 2012).

1.1 RELATED WORKS

1.1.1 Comparative Study of Phonecode

Prechelt (2000) studies 80 implementations of a phonecode program in seven languages. The results show that designing and writing the program in script languages usually takes no more than half as much time as writing in C, C++ and Java.

1.1.2. Comparison of Software Platforms for Web Development

Prechelt (2011) compared different softwares platform for web development in the course of a publicly held 30 hour contest, 9 professional developer teams were given the task of implementing a web application based on the same specification. The languages Java, Perl and PHP were each used by three of the participating teams. The resulting programs were evaluated both on external user relevant and internal developer-related criteria. Notably among the study's findings, the solution quality varied most between the Java teams and least within the PHP group.

1.1.3. Comparative Study of Internet Message Access Protocol (IMAP)

Cesarini *et al.*, (2008) compared implementations of Internet

Message Access Protocol (IMAP) client libraries in five languages Erlang, C#, Java, Python, and Ruby. The libraries differ in functionality, so the comparison includes functionality of primitives as a metric besides lines of code, memory requirements and execution time.

1.1.4. Comparative Study of 10 programming language within 10 Diverse Criteria

Slaiman *et al.*, (2012) compare 10 programming languages within 10 diverse criteria using theoretical features approach implemented in the programming languages and find out the efficiency of the programs without using standard parameter, empirical approach or simulation method.

1.1.5. Comparison of the complexities of tree search algorithms

Aremu and Salako (2006) Present Efficiency Comparison of some selected programming Languages. They addressed how to measure and compared the complexities of tree search algorithms using software metrics approach. The implementation of the algorithms was carried out using C, C++, Pascal, Visual Basic and java programming Languages and they concluded which programming languages is best for breadth first search, depth first algorithms and which programming language is worst for implementing the entire search.

1.1.6. Comparative Study of programming Languages in economic

Boragan *et al.*, (2014) compared programming Languages in economic. They solved the stochastic neoclassical growth model, the workhorse of modern macroeconomics, using C++11, Fortran, java, Julia, python, matlab, mathematica and Ruby by implementing the same algorithms value function iteration with grid search in each of the languages and they reported the execution times of the codes in Mac and Window which are two popular machine for software development economist.

1.1.7. Ongoing Research Projects in Mobile Computing

Arkady and Zahir (2015) found that there has been a considerable research effort going on around the world with respect to mobility. A number of research groups have been set up all around the world to deal with the various aspects of mobility.

2.0 MATERIALS AND METHODS

In this section, we describe the methodology used to obtain the results, including discussion on performance measures for assessing the strength of programming languages, system architecture and design of simulation.

2.1 Performance Measures for Assessing the Strength of Programming Languages

2.1.1 Encryption/Steganography Scheme (Data Security)

The first step, to design encryption and steganography scheme to check the strength of the data security was implemented in the mobile device for the three chosen programming languages (C#, java & vb.net). Encryption using AES-MPK algorithms is based on the Rijndael cipher and steganography is a system of hiding message in an embedded image. Merge encryption and steganography the output become stego image (Schneider, 1996).

The AES-MPK has Four Types of Transformation Bytes that passes through the three programming languages (Marwa *et al.*, 2016).

1. **Substitution byte-** It performed operation in two ways:
 - Multiplicative Inverse
 - Apply Affine Transformation
2. **Shift Rows:** It uses permutation to shift byte in a circular fashion.
3. **Mix columns byte:** This stage, change a bit inside a byte based on bits inside the neighbouring bytes. The reason for these type of transformation is to caused diffusion at the bit level.
4. **AddRound key:** it uses simple bitwise XOR operation with expanded key.

Here, our target is the average execution time that is, the time taken (T) to run it executable statements in C#, java and vb.net for data security in mobile devices.

2.1.2. Download Scheme (Data Connectivity)

The second step was, design download scheme using Client Application to Server Application technique by fetching the numbers of files needed for evaluation, was used to check the strength of Data Connectivity and Data Transfer as well in other way (that is, server application to client Application technique) which was implemented in the mobile device for evaluation of the three programming language (C#, Java & Vb.net). Several elements of the .NET API in window platform can be used for peer to peer application program though in particular, the .NET API has class namespaces that can be used for peer to peer development. System.Net namespace: This namespace has a set of classes integrating Internet protocols into an application. It uses a layered approach for varying levels of control can be added to an application from pure socket control to generate request/response actions in window platforms. The client sends a request to the server via HyperText Transfer Protocol (HTTP). In Java the first technology is JXTA, short for juxtapose - meaning side by side - (<http://www.jxta.org>), an open source API, containing a generalized set of Peer to peer is produced by Sun Microsystems Inc. The intention of these protocols is to allow any device that is available on the network to communicate and collaborate as peers. It is important to note that JXTA is build using Java primitives.

2.1.3. Upload Scheme (Data Transfer)

The Third step, to design upload scheme, which is based on connection of server app - client app check the strength of data Transfer processes in the three languages. The same process which was used in download scheme to fetches files is applied in this process to enable peer to peer communication but in a reverse case. The technique for illustration of Download and Upload Schemes in a Mobile Computing as follows:

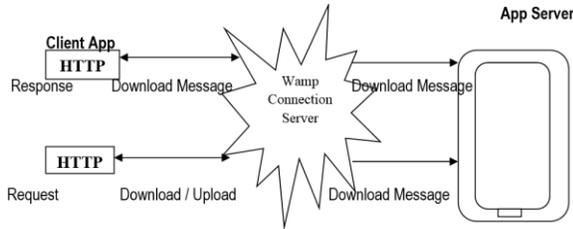


Fig 1: The System Illustration depict Upload and Download Schemes

In this paper, image files of different sizes were stored in the server. The server listens to client requests, searches and processes the required image files segment, and returns it to the client. Upon receiving the requested image file segment, the client displays the image file segment which user trace on the mobile device (android and window phones).

The last objective, after implementation of the above schemes in mobile devices, simulation method was conducted to evaluate the time taken (T) for each criterion and compared.

2.2. SYSTEM ARCHITECTURE

In data security to hide a secret image from android mobile microSD card, encrypt secret image, convert encrypted secret image into stream of binary bit. Read the lower middle of encrypted secret image, byte, and hide lower nibble of channel pixel byte of cover image. Read the upper nibble of encrypted secret image byte into next channel byte of cover image until encrypted secret image is completely hide. Merge an image with hidden encrypted image and send for execution.

In upload and Download schemes, Wamp server messaging is a service that allows to send data from your server to the user's Window powered device. It is a light weight message notifying your application about new data to be fetched from the server like new version of apps. The service serves all aspects of storing, queuing and delivery of messages to the target Window application running on particular device. Wamp server message architecture includes it connection server, an app server in your application that communication with the connection via HTTP protocol and client application in mobile device.

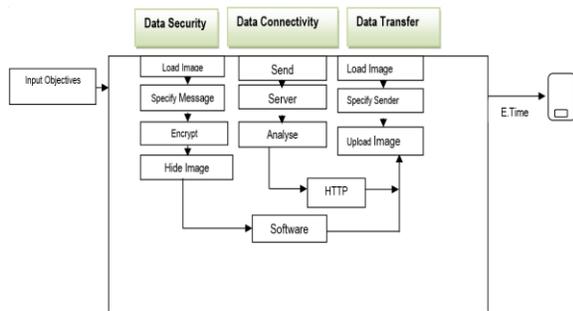


Fig 2: The System Architecture

2.3. Design of Simulation

The mobile device retrieves the number of application processors from the target devices then creates image files on each application and runs for experiments (data security, data connectivity and data transfer) computations in a single loop. Image.jpg is an image file for encryption/steganography that read n-block of data at a time where n is a number of image file sizes

such as 100, 200, 300, 400 and 500 mb (that is the sample images files that was created for the tests), such that the first n-block can be assigned to processor-1 for performing encryption/steganography test, second n-block can be assigned to processor-2, ..., 5th term block. In this case, we are performing encryption/stago scheme on a linear blocks of data by using the concept of linear multiprocessors process. The performance of encryption/steganography scheme is measured by the use of number of counter measure (that is, emulator processor) which was also used in other schemes performance experiment. The time is returned in milliseconds. Encryption/steganography scheme was compared on the three programming languages with each, a sample image files repeated one thousand (1000) times because the resulting times do tend to differ each time the same file is encrypted and embedded in the three languages and five different sample images file size were tested. The fastest execution time out of the three languages as seen in figure (3, 4 & 5) when file size of 100 mb was tested the execution time differ in the three implemented languages. This process will continue till the end of image files and after this experiment in C#, java and vb.net implementation, the average execution time for the entire each programs (files) process were calculated and recorded.



Fig 3: Screen Shot when operation capture information (data security)

Encrypting/ embedding image file 100 mb in mobile devices for running time (T), an android java give 512, C#.net give 143 and vb.net give 183 in milliseconds.



Fig 4: Screen Shot when operation captures information (data transfer)

When a file of about 100 mb undergoes transformation process specifically upload scheme, the average execution time for the android java give 6970 ms, for C#.net give 7593 ms and vb.net give 364 ms respectively.



Fig 5: Screen shot when operations capture information (data connectivity)

Figure 5 Shows implementation of the mobile device application tested with image sample file size 100 mb on the three Programming language for data connectivity.

The average execution time for the android java on download scheme was 9385, for C#.net give 2482 while vb.net execution time 255 all in milliseconds.

3.0 RESULTS AND DISCUSSION

3.1 Results of simulation study

The main tasks of this experiment were collect on the running time, that is the time associated with the schemes (encryption/steganography, upload and download) and the data size imposed by it selected schemes.

Figures 6, 7 and 8 show the results after running executable files for the three chosen programming languages, C#, java and vb.net.

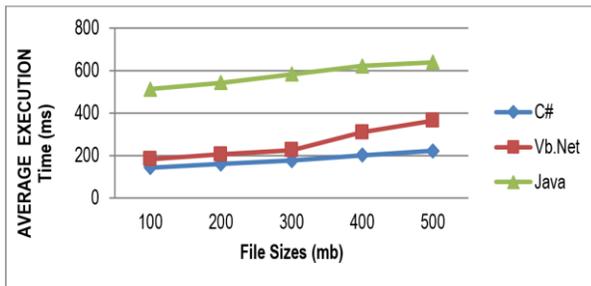


Fig 6: The graphical representation of time for data security (encryption/steganography)

Figure 6 shows graphical representation of time for encryption/steganography process. In this graph blue line shows the encryption and embedded time for C# implementation and the red line shows the encryption and embedded time for vb.net implementation and green line shows the encryption and embedded time for java implementation. Graph shows the difference in execution time for C#, java and vb.net implementation for encryption and embedded process. Here we can see the performance improvement in the C# implementation. In this it can be seen that the performance is not fixed or constant for all file sizes.

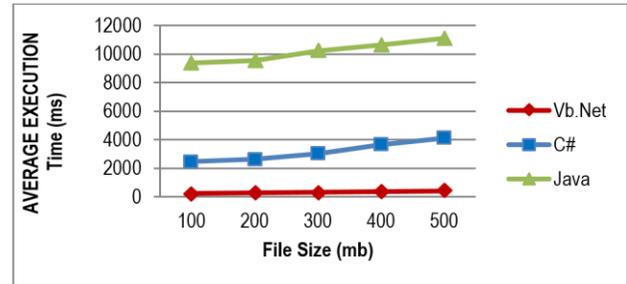


Fig 7: The graphical representation of time for data connectivity (download scheme)

Figure 7 shows graphical representation of time for download scheme process. In this graph, blue line shows the download file (data connectivity) time for C# implementation and the red line shows the download file time for vb.net implementation and the green line shows the download file time for java implementation. Graph shows the difference in execution time for the three languages implementation for download file process. Here we can see the performance improvement in the vb.net implementation. In this also it can be seen that the performance is not fixed or constant for all file sizes. Here we reported that the performance of vb.net implementation for small size file is less and it will increase as the file size will increase but not as the other two programs.

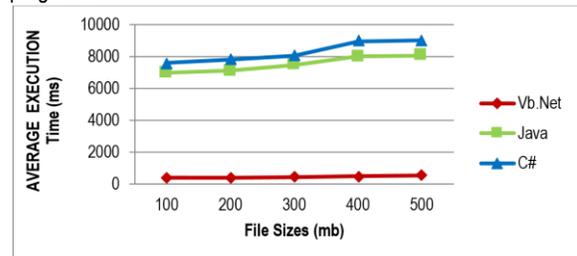


Fig 8: The graphical representation of time for data transfer (upload scheme)

From the above graph (figure 8) the blue line show execution time performance of C#, the red line show performance of vb.net and the green line show the performance of java. Vb.Net is the best in the test of upload file (data transfer). The performance of Vb.Net is much better than other two programming languages. However, upload file has the worst performance in the C# than the java. C# which ranks the last position performance means it spends much more time to execute the statement than other two programming languages.

3.1.1 Discussion of Simulation Results

After carefully analyses the work, the results are compare with other related work. Tables1 was used to compare the results by considering 100 Mb file size from (Zaidman, 2004).

For his experiment seven dummy files, those being files completely filled with the character zero, were created with a file size of 1 Kb, 10 Kb, 100 Kb, 1 Mb, 10 Mb, 50 Mb and 100 Mb. Each application puts the complete file in the system memory, encrypts it and writes it to a file. Then the newly created file was read each time it undergoes the process. The performance of encryption was measured using the number of counter measure (emulator processor) that is also used in the Peer to Peer (file sending and file receiving) performance experiment.

Table 1: Average execution time for symmetric encryption and data transformation

Programming Languages	AES Execution Time (milliseconds)	File Sending Time (milliseconds)	File Receiving Time (milliseconds)
C#	58922	40188	39078
Java	98094	38172	39125
Vb.Net	(Matlab)	1070	27125

Source: Zaidman (2004)

Table 1 shows the average execution time for symmetric encryption and data transformation (file send and receive) obtained in Zaidman (2004) which we used to compare our simulation results. In our simulation results the encryption/steganography scheme for sample of file size 100 mb had a time performance of 143 ms for C#, which better compared with the value of 58922 ms in Zaidman (2004). However, this outrageous value obtained in Zaidman (2004) can be explained by his stop watch which subjected to human error. For the download scheme (data connectivity) for file size 100mb had a time performance of 255 ms and upload scheme (data transfer) has 384 ms for average execution time both for Vb.Net while Zaidman (2004) had a time performance of 40188 ms for sending file and 39078 ms for receiving file both for Vb.Net. Performance of the download and upload of files using HTTP, measured using a number of counter measure (emulator processor) better the performance of Zaidman (2004). Therefore, the data security, download and upload schemes implemented in the mobile device specifically android and window phones enhanced better time performance.

4.0 Summary, Conclusion And Recommendation

4.1 Summary

The study was designed to compare between three programming languages, java, c# and vb.Net based on data security, data transfer and data connectivity. The study was guided by four specific objectives which include to developed a mobile application programs for c#, Java and vb.Net then compared the strength of data security, data transfer and data connectivity with the use of speed approach (running time) to evaluate which programming Language performed best in order to minimize the cost of resources use, energy and maintenance when developing new software. The findings of the study revealed the most suitable application fields for these three of the most popular programming languages according to the average execution time from mobile device and the result from the experiment. C# is suitable for data security. Vb.net is appropriate for the data connectivity such as download scheme. Finally, Vb.net is suitable for data transfer using upload scheme.

When we compare the three programming languages in all the three criteria, it was observed that Android Java had the highest records for both data security and data connectivity over HTTP protocol because it standard has a long footprint causing the execution time to read the files in its memory too long which takes up processor time.

In order to study the performance of the system model of mobile device, a performance analysis for evaluation of programming languages which was implemented on a machine with 2.1 GHz CPU and 4GB RAM along with Ice cream Sandwich 4.0, Windows phone 8.1, software development kit (SDK) and java development kit (JDK) in support.

4.2 Conclusion

We have built three criteria into programming languages, a mobile application programs which provides user experience performance analysis. It can provide objective and stable performance results for variable window devices. Mobile application program can provide user experienced and system level performance value. We evaluated the suite for three programming languages and the results show stable results. After carefully analysed the programming languages, we have come to the conclusion that every languages has its own ups and downs. Every particular languages has a purpose but can be extended or revised to accommodate the current needs of programming languages. In spite of all this, every language has its own specialty and considerably better programming practices which has made it popular and revolutionised the computing world.

4.3 Recommendation

Based on the findings of the study, the following recommendations were made:

- The Application programs implemented in the mobile device should be test with other standard parameters to enhance stable performance.
- The JDK and SDK if installed in the operating system of the machine used improved the work better.
- Academic and ICTS' solution provider in Nigeria should adopt this research under study to put in practice.
- We recommend this system of comparing programming languages should be implemented in mobile phones so that when any students need to check the strengths of any programs at anywhere, at any time base on the certain criteria, these aforementioned methods should be the strategic decision. Also, there will be an open platform to enable more criteria depend on the choice of the programmer.

REFERENCES

- Aremu, P. and F. E. Salako, 2006. Present Efficiency Comparison of some selected programming Language in The second ACM SIGPLAN conference on History of programming languages.
- Arkady Z. and Zahir T. (2015). Mobile Computing: Overview and Current Status. School of Computer Science & Software Engineering, Monash University 900 Dandenong Road, Caulfield East 3145, Australia.
- Boragal A. Arogba P. E, Martin R, Peter G.H (2014). Functional Programming in Economics..Mimeo in preparation. Comparing Solution Methods for Dynamic Equilibrium Economies..Journal of Economic Dynamics and Control 30, 2477-2508
- Cesarini G. Simon L. J, Jasson P. 2008. Cesarini, F., Pappalardo, V. and Santoro, C., "A comparative evaluation of imperative and functional implementations of the imap protocol", in Proceedings of the 7th ACM SIGPLAN workshop on ERLANG, ERLANG '08, ACM, New York, NY, USA (2008).
- Cook, D. (1999) "Evolution of Programming Languages and Why a Language is Not Enough
- Cryptography Algorithms Simulation Based Performance Analysis", International Journal of Emerging Technology and Advanced Engineering, Vol I, Issue 2, December 2011, pp.6-12.

- Chen H. (2010). Comparative Study of C, C++, C# and Java Programming Languages. Vaasan Ammattikorkeakoulu University Of Applied Sciences Degree Program of Information Technology, pp 4-78.
- Marwa E. Abdelmgeid M. and Fatma, O. (2016). Data Security Using Cryptography and Steganography Techniques. International Journal of Advanced Computer Science and Applications, Vol. VII, No. 6, 2016
- Prechelt, L. (2000), An empirical comparison of seven programming languages for a search/string-processing program. [http://www.wip.ira.uka.de/EIR/Technical Report2000-5](http://www.wip.ira.uka.de/EIR/TechnicalReport2000-5) March 10, 2000.
- Prechelt, L. (2011) "A Web Development Platform Comparison by Exploratory Experiment Searching for Emergent Platform Properties", IEEE Transactions on Software Engineering, 37, 1, (2011), 95–108, ISSN 0098-5589, doi:10.1109/TSE.2010.22.
- Schneider, B. (1996): Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd Edition. Wiley and Sons Inc.
- Sleiman R, Rana N. Mohammad F. N, Sheetal H. (2012). Comparative Studies of 10 Programming Languages within 10 Diverse. COMP 6411 - A Comparative studies of programming languages, pp. 1-139.
- Zaidman A. (2004). Comparing file transfer and encryption performance of Java and .NET. Universiteit Antwerpen (UA) Faculteit Wetenschappen Departement Wiskunde – Informatica, pp 10-68.