

# Implementation and Evaluation of User Activity System as a Web Service

Mukesh Rawat, Suryoday Smann

**Abstract**— Web service is a service which allows different systems to connect with each other via World Wide Web (www). It uses technology like Http (Hyper Text Markup Language). It was originally designed for human to machine communication, but now used for machine to machine communication. We have to develop an application that is platform independent and can be availed by anyone from anywhere, it would be only possible if it is platform independent that is there is no restriction of the language which is to be used for running the application as it will be using web services to fetch the data and showing the desired results and we would be showing the comparison between the Standalone and the web based services. Developing file service as web services have many benefits such as interoperability, low cost of communication, supports remote procedure calls, supports document exchange etc. In this thesis file services efficiency measured with the help of performance criteria non as F-measure.

## I. INTRODUCTION

Web administration is an administration which enables various frameworks to interface with one another by means of World Wide Web. It utilizes innovation like Http (Hyper Text Markup Language). It was initially intended for human to machine correspondence, however at this point utilized for machine to machine correspondence. These are web applications which cooperate with other web applications to trade data. Web Services are language independent and communicate by using protocols and data formats etc. These are distributed applications that are available externally. There are two types of web services which are available which are as follows:

- RESTful Web Services:- Collection of web resources identified by URL.
- SOAP Based Web Services:- Used for complicated operations.

It can transform the applications into web based applications. It is a software i.e used over the internet and uses standardized XML system. XML system encodes the data which is being communicated over the internet. Web administrations are dynamic applications that can be distributed, situated to make procedures and supply chains. It isn't limited to just a single language or working framework. The web administrations have the accompanying parts:

- SOAP (Simple Object Access Protocol):- Used for moving the information.
- UDDI (Universal Description, Discovery and Integration):- It indicates the accessibility administrations.

• WDSL (Web Services Description Language):- It officially records which administrations are accessible.

These days, Web administrations are a standout amongst the most significant improvements in the product field which has acquired numerous advantages programming structure and usage. With the development of the Web benefits, countless Web administrations with the equivalent or comparable capacities have been created and discharged. The Web administration choice depends on the Quality of Service (QoS) which has been alluded for taking care of the issues, which is demonstrating the contrast between those Web administrations with a similar capacity and utilizing various arrangements of the QoS levels.

QoS-based administration approaches dependably expect that the QoS information emerging from the specialist co-ops are compelling and worth trusting. In any case, the estimations of QoS qualities that are given by the different specialist organizations might be truly important, since specialist co-ops once in a while may promote higher QoS information than the authentic dimension of the administration so as to draw in an ever increasing number of clients to utilize their administrations and help in increasing more benefits. For instance, the greatest reaction time of the administrations may rise, while the summon rate stays under a specific incentive during the runtime.

## II. GENERAL ARCHITECTURE

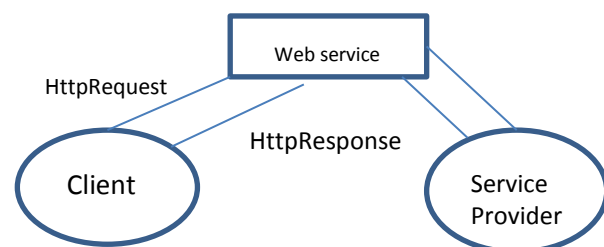


Figure-2.1 General Architecture

Web administrations give a standard methods for utilizing these administrations among the distinctive programming applications, running on an assortment of stages which implies that the application isn't constrained to just a single stage. Regular meaning of a Web administration utilizing WSA (Web Service Architecture), and characterize its place among the bigger Web administrations system to support the network. WSA gives a model and aides in understanding the Web administrations and the connections which are in the middle of the different determinations and innovations that

Revised Manuscript Received on July 18, 2019.

Mukesh Rawat, Dept. of CSE, MIET, Meerut, Uttarpradesh, India.  
(E-mail: mukesh.rawat@miet.ac.in)

Suryoday Smann, Dept. of CSE, Translam Institute of Engg. & Technology, Meerut, Uttarpradesh, India. (E-mail: suryoday.smann@gmail.com)

comprise of the WSA. WSA advances interoperability through the meaning of perfect conventions.

The above architecture states that a client has sent a request to the server, now this request will pass through a web service interface that actually sends the traffic ahead to the server where the request gets processed according to the response and after the processing is done the response is sent back to the web service and then the web service sends the response to the client. This process takes a very short span of time and the processing speed is also fast. The better the internet connection is the more accurate the results and easily we get.

This paper proposed building a web service which through which the codes that are written in java and dot net will access the same web service and will show that platform dependency can be removed and we can access our code at anywhere, any place we want to. This has made the ease to access the information as we do not have to search for either net beans or the visual studio it can be simply accessed by the web service that is running on the server side and the client sends a request which is processed by the web service interface and sent to the desired location and the response is generated according to the request.

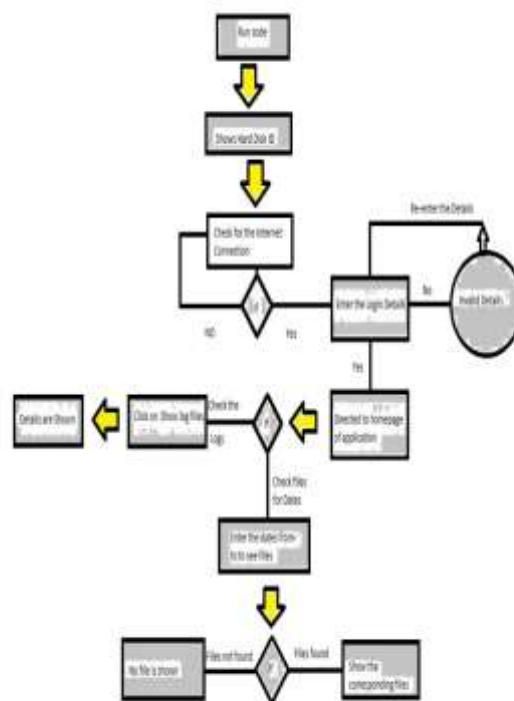
My thesis here is accessing the drives on a system and can access them anywhere in the world. In this when we run the code it simply ask for a login details, after filling out the required details and an internet connection will be mandatory as web service is running over the internet and without it, the information cannot be accessed. When the code is run for the first time it registers the hard disk id and makes a configuration file and saves it. When we login it shows the hard disk id of the system which is registered at that particular period of time and then we receive the main page of the service. Here we can see all the drives present on the system and can access them here only and even we can make changes like we have given three features like **unzip, delete and compress**. These functions can be performed and will be reflected in the system and even we have the log details as it shows when did the admin logged in and at what time he logged out. It also shows the hard disk id in one column, in other it shows the date/time that is when the function was performed and to keep a record of all what is being done while accessing the service and the last column shows what action has been performed. For example we selected drive D and opened up a folder containing a document file, now we want to zip it so we will right click on it and click on the zip button which has been provided. Now the time and date at which the function is performed and stored into the log details and we know at what time the action was performed.

It also has the functionality which is the main reason why we are here discussing it as there is option of selecting the dates i.e from – to, it means that suppose we enter dates like 13-jun2015 to 25-april-2017. So it would only show the files which were present during this period or have been modified or any other action has been performed on them. By this we will be easily locating our data if we know about the details of the date as it would require less effort than searching for it manually. We will be seeing the input and the output screens as we move further and discuss in detail about the various aspects of the thesis. We can also look for the log

details if we want to as it is having a button when clicked on it shows the details of the action been performed. As code is in both java and dot net we have single web service which is accessed by both and in the log details or even while checking the dates the each and every record is shown either performed by using the code in dot net or in java. Each and every log detail can be seen and the action performed. Now we will see the login page for both languages i.e JAVA and DotNet. The first we will be discussing is the java as you can see the snapshots of how the login page looks like. Firstly we need to open the application in Netbeans and select open application from the file menu and open the application. Now we will right click on the form login page and select run file. Now we will see the pop up message saying it is using web service and must check for the internet connection as it is mandatory for the thesis to because the web service will only be accessed if there is a net connection available.

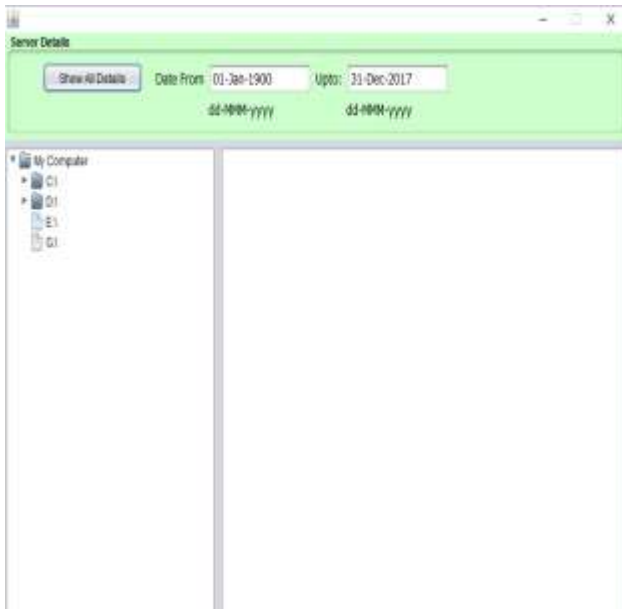
Below we have the figure which explains how the process is initiated and what is done in the next step and we will find out whether the request given is to be initiated further or it gives an error to check the details mentioned. Clearly, we will see in the figure below and will explain how the process flows through a set of patterns and find out the result we are looking forward to.

*Process Flow*



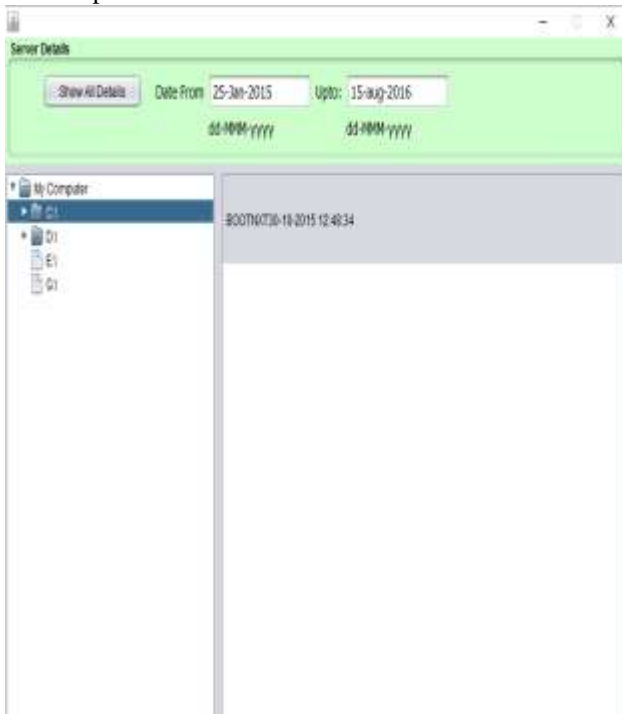
**Figure – 2.2 working of the application**

### III. INPUT AND OUTPUT SCREENS



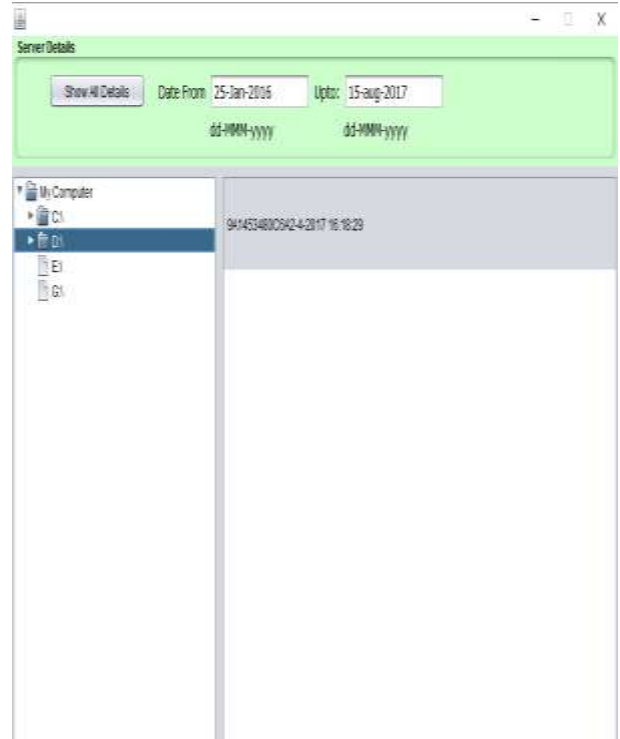
**Figure-3.1 Homepage of Java Application**

In the above figure we can see that there are dates from and to, in this we can put the dates from which we want to see the files. For example if we want to see the files which are present between 25-june-2016 to 15-aug-2016. The output result will show the files which were there during the same time period and can be seen as follows:



**Figure-3.2 Show files between two dates**

Here we can see the file with the date and time has been displayed as per the request was given.



**Figure-3.3 Another Example**

Here is the second example for the above said. We also can see the log details that what operations have been performed like login, logout, compress, unzip and delete and these operations can be performed by right clicking on the file and selecting the particular option that we require to perform as illustrated in the below figure.



**Figure-3.10 Files in the D drive**

In the above figure we have compressed the file winPEge.sys and below it we can see the zip file of the same, it is done by right clicking on the file and selecting the zip option. There are two other functions available as delete and unzip. So all in all we have three available functions in the application.

IV. DISCUSSION ON RESULTS

Quality of Service (QoS) of various activities that are performed and watch that the information originating from the specialist organizations are successful and reliable or not.

Suppose we take the data is productive and reliable then the values of QoS which have been provided are based on the assumptions and in it considers to have an ideal environment, but in the real world we see it does not have the ideal conditions and there is always a difference between the ideal condition evaluation and the real time environment conditions result.

4.1 WS-QoS Measurement Tool

It is intended to get the QoS esteems for a given web administrations. Initially, the WDSL record is obtained from UDDI. Presently the WDSL document is parsed to get administration related data, and the information from the test is produced for each information component. Furthermore, the Web administration classes are created as Java documents by utilizing the WSDL2Java apparatus from the Axis, which gives the administration invoker all the uncovered techniques and parameters. In the third step, the Web administration invoker gathers the produced test date to code to cause the Web administration to be summoned and its reaction results and status to be gathered which can be utilized to process the QoS run of the mill parameters, for example, accessibility, unwavering quality, and availability. In the last advance, time Aspect code weaves time estimation codes when the byte code of the Web administration summoning strategy; at that point the begin time and end time of Web administration call are obtained, and the Web administration reaction time is figured.

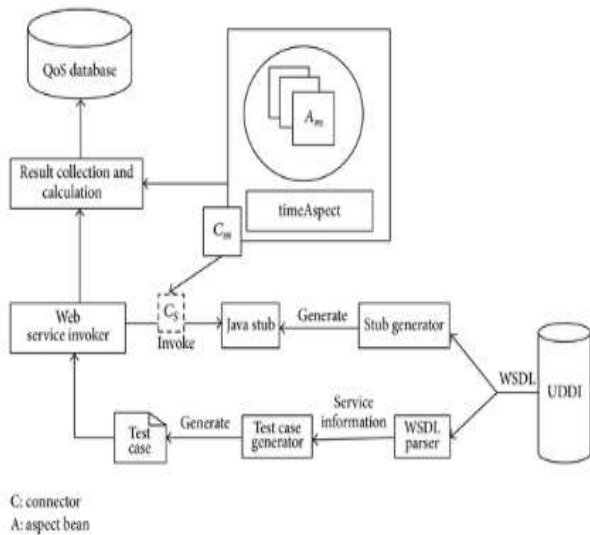


Figure - 4.2 WS-QoS Architecture

4.2 Calculation of the Global QoS

Every call to the web service by WS-QoS tool, the collected Web services QoS values may be different to reflect the real dynamic changes of QoS, the global QoS values of the service must be recalculated and stored on the basis of the historical data and the current data which is available.

For a given a Web service, it has m QoS attributes which can be expressed as  $A: \{A_j, 1 \leq j \leq m\}$ . The Web service QoS attribute values are defined as  $Q: \{q_j, 1 \leq j \leq m\}$ ;  $q_j$  is the value of the attribute  $A_j$ . The  $n^{th}$  current factual QoS data which is collected by the WS-QoS measurement tool is defined by the set;

$fa\_Q_n: \{fa\_q_{nj}, 1 \leq j \leq m\}$ ;  $fa\_q_{nj}$  the  $n^{th}$  factual value of the attribute  $A_j$ . The global QoS value of the service can be calculated by using a two-time average method. In this method we randomly sample  $p$  ( $p < n$ ) numeric from the  $n$  values ( $p \leq n$ ); the  $l^{th}$  QoS value  $fa\_q_{lj}$  can be calculated as shown in formula(1), so by doing sampling  $k$  times we can get  $k$  sampling values, and then the global QoS value of the service,  $fa\_Q_g = \{fa\_q_j, 1 \leq j \leq m\}$ ,  $fa\_q_j$  can be calculated as shown in formula (2):

$$fa\_q_j = \frac{1}{p} \sum_{i=1}^p fa\_q_{ij}, \quad p \neq 0 \quad (1)$$

$$fa\_q_j = \frac{1}{k} \sum_{i=1}^k fa\_q_{ij}, \quad k \neq 0 \quad (2)$$

4.3 Calculation Of the Quality Of Service (QoS) Similarity

Similitude is accomplished by computing the amassed normal of the examination result between the promoted qualities and the true worldwide quality qualities. QoS quality holds two distinct headings or inclinations of their qualities; if the result of the property is certain, it implies that the greater worth is the better one. Despite what might be expected if the result is gotten as negative, it implies that the littler qualities are liked. For instance, for property "reaction time" the littler worth is liked, so the inclination of this parameter is negative, though for quality "accessibility" the greater the worth methods accessibility of items is more, so the propensity is sure. In light of the bearing of the quality, the closeness can be processed by the accompanying recipes (3)– (5). As portrayed over, the worldwide QoS estimation of the administration is  $fa\_Q_g$  and  $fa\_Q_g = \{fa\_q_j, 1 \leq j \leq m\}$ ; the promoted QoS which demonstrates the quality being offered by the Web specialist co-op is characterized by the set  $ad\_Q = \{(ad\_min\_q_j, ad\_max\_q_j), 1 \leq j \leq m\}$ .  $ad\_min\_q_j$  and  $ad\_max\_q_j$  alludes to the base worth and the most extreme estimation of the publicized quality trait separately. Think about the accompanying condition:

$$Sim = \sum_{j=1}^m c/m \quad (3)$$

Now if the tendency of the attribute is negative then,

$$c = \begin{cases} 0, & \text{if } fa\_q \leq ad\_max\_q \\ 1, & \text{if } fa\_q \geq ad\_max\_q \end{cases} \quad (4)$$

$$c = \begin{cases} 0, & \text{if } fa\_q \leq ad\_min\_q \\ 1, & \text{if } fa\_q \geq ad\_min\_q \end{cases} \quad (5)$$

4.4 Calculation Based on the Date

The following table is created to calcite the reputation level of the

Here  $Q_1, Q_2, \dots, Q_{10}$  are the request which have been made by the user against a particular response. In the below table we have the request made by user, response time that is the time taken by the web service to respond to the request, number of accurate results and accessibility.

<u>t</u>	<u>Reques</u> <u>e Time</u>	<u>No. Of</u> <u>Accurate</u> <u>Results</u>	<u>Accessibilit</u> <u>y</u>
Q <sub>1</sub>	0.3	20	0.05
Q <sub>2</sub>	0.2	22	0.07
Q <sub>3</sub>	0.04	10	0.03
Q <sub>4</sub>	0.06	18	0.09
Q <sub>5</sub>	0.07	12	0.01
Q <sub>6</sub>	0.5	19	0.02
Q <sub>7</sub>	0.6	16	0.03
Q <sub>8</sub>	0.8	28	0.04
Q <sub>9</sub>	0.5	27	0.06
Q <sub>10</sub>	0.32	25	0.07

**Table- 4.1 Based On Date**

Here, we will make the pairs and find the difference between the factual values and the advertised values and see whether the difference comes out is small or large. If the difference is large then we would say that the advertised values are not good and have not been properly taken into consideration whereas if the difference come out to be small then the value.

To calculate the values we will make pairs of the findings in the table above

- Q<sub>1</sub> = { .3, 20, .05}
- Q<sub>2</sub> = { .2, 22, .07}
- Q<sub>3</sub> = { .04, 10, .03}
- Q<sub>4</sub> = { .03, 18, .09}
- Q<sub>5</sub> = { .07, 12, .01}
- Q<sub>6</sub> = { .5, 19, .02}
- Q<sub>7</sub> = { .6, 16, .03}
- Q<sub>8</sub> = { .8, 28, .04}
- Q<sub>9</sub> = { .5, 25, .06}
- Q<sub>10</sub> = { .32, 25, .07}

**For Q<sub>1</sub>**

**QoS by formula (1) .....**

$$fa_{qj} = \frac{1}{p} \sum_{i=1}^p fa_{qij}, \quad p \neq 0$$

if p=2, then

$$fa_{q1} = \frac{1}{2} (.05 + 20) = 10.025$$

Sampling values (k) = 3

For k1, p=2, values of Q<sub>1</sub> = { .3, 20}  
= 1/2 (.3+20)

Formula

[1] = 10.15

For k2, p=2, values of Q<sub>1</sub> = { 20, .05}  
= 1/2 (20+.05)

Formula

[1] = 10.025

For k3, p=2, values of Q<sub>1</sub> = { .3, .05}  
= 1/2 (.3+.05)  
= .175

Formula [1]

**QoS by formula (2) .....**

$$fa_{qj} = 1/k \sum_{i=1}^k fa_{qij} = 1/3 (.3 + 20 + .05)$$

= 6.78

**For Q<sub>2</sub>**

**QoS by formula (1) .....**

$$fa_{qj} = \frac{1}{p} \sum_{i=1}^p fa_{qij}, \quad p \neq 0$$

if p=2, then

$$fa_{q2} = \frac{1}{2} (.2 + 22) = 11.1$$

Sampling values (k) = 3

For k1, p=2, values of Q<sub>2</sub> = { .07, 22}  
= 1/2 (.07+22)

Formula

[1] = 11.035

For k2, p=2, values of Q<sub>2</sub> = { 22, .2}  
= 1/2 (20+.2)

Formula

[1] = 11.1

For k3, p=2, values of Q<sub>2</sub> = { .2, .07}  
= 1/2 (.3+.05)

Formula

[1] = .135

**QoS by formula (2) .....**

$$fa_{qj} = 1/k \sum_{i=1}^k fa_{qij} = 1/3 (.2 + 22 + .07) = 7.42$$

**For Q<sub>3</sub>**

**QoS by formula (1) .....**

$$fa_{qj} = \frac{1}{p} \sum_{i=1}^p fa_{qij}, \quad p \neq 0$$

if p=2, then

$$fa_{q3} = \frac{1}{2} (.04 + 10) = 5.02$$

Sampling values (k) = 3

For k1, p=2, values of Q<sub>3</sub> = { .03, 10}  
= 1/2 (.03+10)  
= 5.015

Formula [1]

For k2, p=2, values of Q<sub>3</sub> = { 10, .04}  
= 1/2 (10+.04)  
= 5.02

Formula [1]

For k3, p=2, values of Q<sub>3</sub> = { .04, .03}  
= 1/2 (.04+.03)  
= .035

Formula [1]

**QoS by formula (2) .....**

$$fa_{qj} = 1/k \sum_{i=1}^k fa_{qij} = 1/3 (.04 + 10 + .03) = 3.35$$

Similarly, we can calculate the rest values by following the above technique and reach to the desired result.

**Calculation Based on Logs**



	<u>Response Time</u>	<u>No. Of Accurate Results</u>	<u>Processing Time</u>
K <sub>1</sub>	0.2	1	0.05
K <sub>2</sub>	0.04	1	0.07
K <sub>3</sub>	0.07	1	0.03
K <sub>4</sub>	0.03	1	0.09
K <sub>5</sub>	0.1	1	0.01

Table – 4.2 Based On logs

- K<sub>1</sub> = { .2,1.05 }
- K<sub>2</sub> = { .04,1.07 }
- K<sub>3</sub> = { .07,1.03 }
- K<sub>4</sub> = { .03,1.09 }
- K<sub>5</sub> = { .1,1.01 }
- K<sub>6</sub> = { .08,1.02 }
- K<sub>7</sub> = { .06,1.03 }
- K<sub>8</sub> = { .08,1.04 }
- K<sub>9</sub> = { .09,1.06 }
- K<sub>10</sub> = { .3,1.07 }

**For K<sub>1</sub>**

**QoS by formula (1) .....**

$$fa_{qj} = \frac{1}{p} \sum_{i=1}^p fa_{qij}, \quad p \neq 0$$

if p=2, then

$$fa_{qj} = \frac{1}{2} (.1 + 1) = 0.55$$

Sampling values (k) = 3

For k1, p=2, values of K<sub>1</sub> = { .1, 1 }  
 $= \frac{1}{2} (.1+1)$   
 $= 0.55$

Formula [1]

For k2, p=2, values of K<sub>1</sub> = { 1, .05 }  
 $= \frac{1}{2} (1+.05)$   
 $= 0.525$

Formula [1]

For k3, p=2, values of K<sub>1</sub> = { .1, .05 }  
 $= \frac{1}{2} (.1+.05)$   
 $= 0.075$

Formula [1]

**QoS by formula (2) .....**

$$fa_{qj} = 1/k \sum_{i=1}^k fa_{qij} = 1/3 (.1 + 1 + .05) = 0.383$$

**For K<sub>2</sub>**

**QoS by formula (1) .....**

$$fa_{qj} = \frac{1}{p} \sum_{i=1}^p fa_{qij}, \quad p \neq 0$$

if p=2, then

$$fa_{qj} = \frac{1}{2} (.04 + 1) = 0.52$$

Sampling values (k) = 3

For k1, p=2, values of K<sub>2</sub> = { .04, 1 }  
 $= \frac{1}{2} (.04+1)$   
 $= 0.52$

Formula [1]

For k2, p=2, values of K<sub>2</sub> = { 1, .07 }  
 $= \frac{1}{2} (1+.07)$   
 $= 0.535$

Formula [1]

For k3, p=2, values of K<sub>2</sub> = { .04, .07 }  
 $= \frac{1}{2} (.04+.07)$   
 $= 0.055$

Formula [1]

**QoS by formula (2) .....**

$$fa_{qj} = 1/k \sum_{i=1}^k fa_{qij} = 1/3 (.04+ 1 + .07) = 0.37$$

**For K<sub>3</sub>**

**QoS by formula (1) .....**

$$fa_{qj} = \frac{1}{p} \sum_{i=1}^p fa_{qij}, \quad p \neq 0$$

if p=2, then

$$fa_{qj} = \frac{1}{2} (.07 + 1) = 0.535$$

Sampling values (k) = 3

For k1, p=2, values of K<sub>3</sub> = { .07, 1 }  
 $= \frac{1}{2} (.07+1)$

Formula

[1]  $= 0.535$

For k2, p=2, values of K<sub>2</sub> = { 1, .03 }  
 $= \frac{1}{2} (1+.03)$

Formula

[1]  $= 0.515$

For k3, p=2, values of K<sub>2</sub> = { .07, .03 }  
 $= \frac{1}{2} (.07+.03)$

Formula

[1]  $= 0.05$

**QoS by formula (2) .....**

$$fa_{qj} = 1/k \sum_{i=1}^k fa_{qij} = 1/3 (.07+ 1 + .03) = 0.366$$

Similarly, we can calculate the rest values by following the above technique and reach to the desired result.

**V. CONCLUSION**

Web administration is an administration which enables various frameworks to associate with one another by means of World Wide Web. It utilizes innovation like Http (Hyper Text Markup Language). It was initially intended for human to machine correspondence, however at this point utilized for machine to machine correspondence. These are web applications which associate with other web applications to trade data. Web Services are language autonomous and convey by utilizing conventions and information designs and so forth.

This application helps in working on a single application which is platform independent and easy to use. It also has functions like compress, delete and unzip which can be performed and even we can check the log details of the application that has been developed as we can see the login time, logout time and even what operations are performed on the files in the system so as to keep a check who



performed what functions and at what time. We also can see the files by just entering the date from which till to the date we want to see. We calculated the factual values against the advertised values and able to know that when the difference between the factual and advertised values is less than it is said to be effective or we can say that the advertised values were good. On the other hand, if the difference between the factual and advertised values is large then the value is said to less effective. It could be developed for the android mobile phones as it is available with the windows and other operating system. We can improve the web services response time. As it is available for certain languages further it can extend its approach towards other languages also.

## REFERENCES

1. Louridas, P. "SOAP and Web Services Software", IEEE Volume: 23, Issue: 6 Digital Object Identifier: 10.1109/MS.2006.172 Publication Year: 2006, Page(s): 62 – 67.
2. B. Medjahed, A. Bouguettaya, and A. Elmagarmid, "Composing Web Services on the Semantic Web," J. Very Large Databases, vol. 12, no. 4, Nov. 2003, pp. 333–351.
3. David Booth, W3C Fellow (2013, March) Web Services Architecture. [Online]. [www.w3.org](http://www.w3.org).
4. P. Hamill, "Unit Testing Web Services," Dr. Dobb's Journal, vol. 33, pp. 53-58, 2008.
5. S. Sakr, "XML compression techniques: A survey and comparison," Computer and System Science, vol. 75, no. 5, pp. 303–322, Aug. 2009.
6. S. Heinzl, M. Mathes and B.M. Freisleben, "A Web service communication policy for describing nonstandard application requirements," in Proc. Int. Sym. Applications and the Internet 2008, pp. 40–47.
7. F. Curbera et al., "Unraveling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI," IEEE Internet Computing, vol. 6, no. 2, 2002, pp. 86–93.
8. Benslimane, D.; Dustdar, S.; Sheth, A. (2008). "Services Mashups: The New Generation of Web Applications". *IEEE Internet Computing*. **10** (5): 13–15. [www.w3.org/TR/ws-arch](http://www.w3.org/TR/ws-arch).
9. W3C Note, Web Services Description Language (WSDL) 1.1, 15 March 2001, <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>.