Investigating the Suitability of Wireless Web Services to Support Business Processes

Roopesh Kevin SUNGKUR
Department of Computer Science, University of Mauritius, Mauritius
r.sungkur@uom.ac.mu

ABSTRACT

Nowadays we need to access information anytime from anywhere. This greatly increases our productivity and our quality of life. Previously web services were physically tied up to servers but wireless web services have proved to be very convenient and this convenience has been the most important factor behind the adoption of wireless technology. An example to demonstrate this might be the use of a wireless web service in a supermarket to get useful information and this is achieved by using an easy, client-side wireless web service to get the core web services required.
This paper contains a description of an ad-hoc wireless web service that has been developed for a supermarket and will investigate upon the suitability of web services for implementing ad-hoc mobile application integration. Basically when a customer equipped with a Wi-Fi enabled device enters a region of the supermarket within range of wireless access points, he/she will be offered a list of services that are offered by the supermarket. Some of these services might include information about products at a discounted price, queries about particular products, marketing of new products being offered, location of particular items on shelves, identification of new point of sales just opened, just to name a few. This can be potentially beneficial especially considering the number of persons now having smart phones and mobile devices.

KEYWORDS

Web Services, Wireless, Availability, XML, Interoperability

1 INTRODUCTION

Very often only core wireless web services are selectively being requested so as to minimize bandwidth and CPU usage. Applications that utilize a wireless front end for pervasive human interfaces and Web services on the back end to leverage the Internet's vast information resources are called wireless Web Services applications. Web services are software applications that can be described and accessed based on XML and standard Web protocols over intranets, extranets and the internet. Built on XML, a standard that is accepted by the majority of vendors, Web services first focuses on interoperability. Web services technologies such as Universal Description Discovery and Integration (UDDI) allow applications to dynamically discover information about web services. The message syntax for a web service is described in WSDL, the Web Service Definition Language. SOAP, developed as the Simple Object Access Protocol, is the XML-based message protocol for communicating with Web services [1]. The benefits that can be derived from Web Services are numerous and varied, some of which include interoperability, simplicity, lower cost of integration, reusability, availability and scalability of applications developed.

2 LITERATURE REVIEW

Web Services Architecture

The Web Service Architecture has three main components as shown in Figure 1 below

- **A Service Provider**: a service provider is normally the organization that develops the web service. A service provider is also responsible for creating and hosting the service descriptions.
- **A Service Registry:** A service registry is analogous to the yellow pages. It is essentially a publicly searchable information exchange mechanism, in which service providers register and publish their company information and web service descriptions. A service

requestor can search the registry to find the desired web service.

- **A Service Requester:** A service requestor is an application, Web Service, or even a human who wants to use a particular web service.

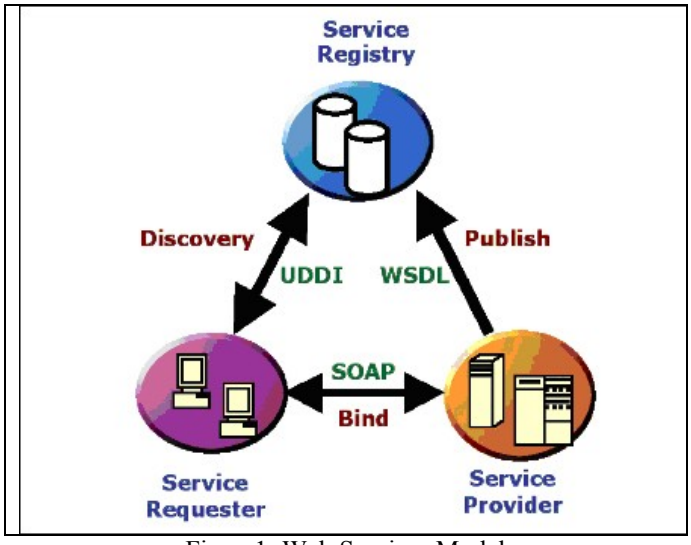

Figure1: Web Services Model

The adoption of Web services standards such as SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) and UDDI (Universal Discovery Description and Integration) help to improve the interoperability of wireless Web services across a variety of network and device infrastructures.

Simple Object Access Protocol (SOAP)

SOAP is an XML-based protocol that allows programs running on heterogeneous hardware and software in a distributed environment to exchange data with each other, and to do remote procedure calls (RPC).

The protocol consists of three parts:

- An envelope, which is an XML document that defines a framework for describing the message.
- A set of data encoding rules
- A convention for representing remote procedure calls and responses.

An example of a simple SOAP message is displayed below

```
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI">
      <symbol>BORL</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Figure2: SOAP Message

Universal Description, Discovery and Integration standard (UDDI)

UDDI is the technology for publishing and finding information about web services [2].
Technically, UDDI specifications consist of two things:

- An XML schema for SOAP messages that provides the necessary data structure to store the different kinds of information.
- UDDI API specification: These APIs enable programmers to publish and find information about web services in a registry. The applications use SOAP messages to both find and publish data.

Web Services Description Language (WSDL)

A web service is useless unless others can find out what it does and how to call it. Developers must know enough information about a web service so they can write a client program that calls it. WSDL is an XML-based language used to define web services and describe how to access them. Specifically, it describes the data and message contracts a web service offers. By examining a web service's WSDL document, developers know what methods are available and how to call them using the proper parameters [3].

WSDL is an XML document that describes a web service, specifies its location and describes the interfaces and methods (called ports and operations, in WSDL-speak) that can be executed on the web service. The role of WSDL is similar to that of Interface Definition Language (IDL) in the CORBA architecture [4].

```
- <definitions targetNamespace="http://Services/" name="Discou
  - <types>
    - <xsd:schema>
        <xsd:import namespace="http://Services/"
        schemaLocation="http://vciltemp.vciltcentral:8080/Superm
      </xsd:schema>
    </types>
  - <message name="getDiscountList">
      <part name="parameters" element="tns:getDiscountList"/>
    </message>
  - <message name="getDiscountListResponse">
      <part name="parameters" element="tns:getDiscountListRes
    </message>
  - <portType name="DiscountService">
    - <operation name="getDiscountList">
        <input message="tns:getDiscountList"/>
        <output message="tns:getDiscountListResponse"/>
      </operation>
```

Figure3: WSDL

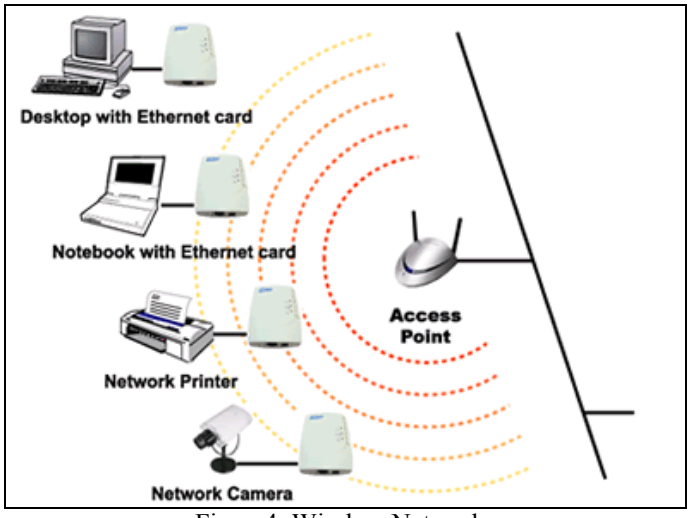

Figure4: Wireless Networks

Ad-hoc Networks

Ad-hoc networks are those that can be formed temporarily while particular devices are in the vicinity of each other. Ad-hoc networks are useful in many application environments and do not need any infrastructure support. Collaborative computing and communications can be set up using ad-hoc networking technologies. So, the typical ad-hoc network is set up for a limited period of time. The application may be mobile and the environment may change dynamically. Consequently, the ad hoc protocols must self-configure to adjust to environment, traffic and mission changes. What emerges from these characteristics is the vision of an extremely flexible, malleable and yet robust and formidable network architecture.

Wireless Networks

The term wireless network refers to any type of network that is wireless whose interconnection of nodes is implemented without the use of wires.

Wi-Fi

Wi-Fi is a wireless technology brand owned by the Wi-Fi Alliance intended to improve the interoperability of wireless local area network products based on the IEEE 802.11 standards.

Wi-Fi is a commonly used wireless network in computer systems which enable connection to the internet or other machines that have Wi-Fi functionalities. Wi-Fi networks broadcast radio waves that can be picked up by Wi-Fi receivers that are attached to different computers.

Bluetooth

Bluetooth is a wireless technology that allows computers, phones and other devices to talk to each other over short distances (typically about 10 meters). Bluetooth uses radio waves (in the 2.4 Gigahertz range), and is designed to be a secure and inexpensive way of connecting and exchanging information between devices without wires.

Architectures for wireless web service

The term wireless network refers to any type of network that is wireless whose interconnection of nodes is implemented without the use of wires.

A wireless web service can feature one of the following architectures:

- **Wireless Portal Network**

In this type of architecture, communication happens via a WAP gateway most of the time and there is no need for the capability of hosting Web Services client code at the mobile device. The request to be sent is embedded in a Wireless Markup Language (WML) message and when the portal receives the message, it is translated into a SOAP request [5]. Once the portal receives the response from the "real" Web Service, it translates the result back into WML and the result is sent back to the mobile device.

Advantage of this architecture:

- Low setup effort
- Minimal processing at the mobile device

Disadvantage of this architecture:

- Requires a central authority with control over user data
- Central point of failure
- Message-level security cannot be supported

- **Wireless Extended Internet**

Wireless extended Internet is the wired Internet's expansion to wireless devices. Today's handheld devices have more computing power than many desktop PCs did 10 years ago. Wireless information devices can have their own IP addresses (through Internet Protocol 6) and full network functionalities. Those devices usually run smart, fat clients that interact with multiple backend services simultaneously and store/process application data on the device [6].

This allows the mobile device to act as a real SOAP Web Service client. All communication is XML-over-HTTP and a TCP/IP connection can be set-up from the mobile device.

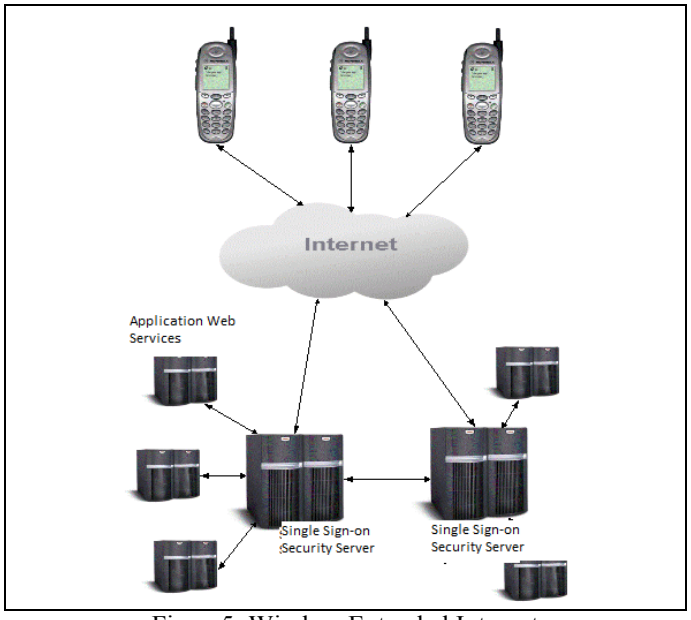

Figure5: Wireless Extended Internet

- **Peer-to-peer**

This allows a mobile device to itself host a Web Service [7].

J2EE v/s Microsoft .NET

The point of Web Services is to make it easier for applications to work cooperatively with other applications regardless of what programming language was used in creating the applications. Web services is not about which programming language or application development is right or wrong but rather the notion of cross platform and program-to-program interoperability should be considered[8].

But nevertheless it is useful to mention some of the arguments for and against using the .NET and J2EE platform [9].

Arguments for .NET and against J2EE

- .NET has an awesome tool story with Visual Studio.NET
- .NET has a simpler programming model, enabling rank-and-file developers to be productive.
- .NET provides for language neutrality.

- .NET benefits from being strongly interweaved with the underlying operating system.

Arguments for J2EE and against .NET

- J2EE is a more advanced programming model, appropriate for well-trained developers who want to build more advanced object models and take advantage of performance features.
- J2EE gives you platform neutrality, including Windows. Developers can use the environment they are most productive in.
- J2EE has a better legacy integration story through the Java Connector Architecture (JCA) [10].

3 METHODOLOGY/PROPOSED SOLUTION

The scenario depicted below will help better understand the process. When a customer equipped with a Wi-fi enabled device, most probably a Personal Digital Assistant (PDA) enters a region of the supermarket within range of wireless access points, he will offered a list of services that are offered by the supermarket.

Some of these services might include:

- Information about products that are at a discounted price.
- Queries about particular products.
- Marketing of new products being offered.
- Location of particular goods/products on shelves.
- Identification of new point of sales terminal that have just been opened (optional).

Of course, all of the functionalities will not be implemented but the most essential ones will be implemented to test the performance of the web service over a wireless infrastructure.

A timer has been implemented to measure the time taken between the client issuing the request and the client receiving the requested information. This will return the time taken for the request in millisecond

The methodology that will be used to develop the system will be incremental prototyping. The reason why prototyping has been chosen as a method for development is that we are not sure how a web service would operate over a wireless infrastructure and it would be pointless to develop a whole system at once when we are not sure that it would function as it should. A prototype with limited functionality will be initially developed. This will basically just provide a core service to the mobile client and then the other services will be incrementally added.
The choice of Wi-fi (802.11) as technology has been due to the fact that it can cover a range of 100 to 150 meters and supports wireless networking. Bluetooth also supports wireless networking but it has lower range (10 meters). Bluetooth also has a much slower speed as compared to Wi-fi. The basic architecture consists of mobile devices communicating with a server at wireless access point when in range.

The adoption of Web services standards such as SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) and UDDI (Universal Discovery Description and Integration) help to improve the interoperability of wireless Web services across a variety of network and device infrastructures [11, 12].

Architecture Chosen for Wireless Web Service

The adoption of Web services standards such as SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language) and UDDI (Universal Discovery Description and Integration) help to improve the interoperability of wireless Web services across a variety of network and device infrastructures [13, 14].
Taking into consideration the advantages and disadvantages of each of these architectures mentioned above and taking into consideration the requirements of the project, the architecture that would best suit our purpose here is the “Wireless Extended Internet”.

This allows the mobile device to act as a real SOAP Web Service client. All communication is XML-over-HTTP and a TCP/IP connection can be set-up from the mobile device. The client program will be more like a fat client.

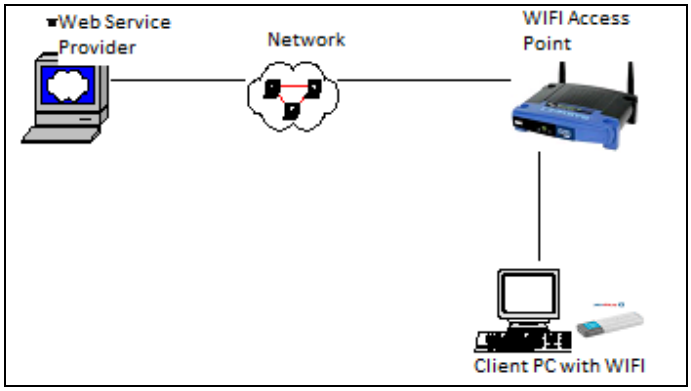

Figure6: Choice of Architecture

Core Technologies Used

Web services are built around standard XML communication protocols. In wireless Web services applications, servers and clients are loosely coupled with XML messages. So both Web services application servers and wireless devices must support standard XML protocols.

On the server side, the J2SE/J2EE platform offers excellent support for generic and Web services-specific XML processing. Core Java technologies are needed to support the described wireless web services and it is believed that Java would be suitable for developing a wireless web service application.

4 RESULTS AND INTERPRETATION

A sample of the returned results is shown below. This is for the first prototype, which shows the articles that are at a discounted price. Appropriate formatting of the results had not yet been done in the first prototype and what mattered more were the results obtained.

Figure7: Web Service Results

The time taken for the Web Service request is also shown in millisecond (13641 ms).

The prototypes that have been developed will be tested and subject to different tests to basically monitor/show the following:

- Comparing the performance of wireless web service as compared to a "traditional" wired web service.
- Investigating whether web services operate suitably in a wireless environment.
- Investigating incremental prototyping as a means of developing wireless web services.
- Availability of the wireless web service developed.
- Usability of the wireless web service developed.

The following performance benchmarks will be investigated for the purpose of measuring the success of the aims proposed:

Availability

- The system should be able to provide service required a large percentage of the time.

Usability

- The length of time it takes to request for a particular web service can be recorded and analyzed. A set of users will be asked to test the interface and to give their feedback and suggested improvements.

Finally a comparison will be made between the conventional way of using web services ("fixed-wired environment") and the wireless one.

However it should be borne in mind that wireless networks have lower bandwidth and higher latency.

Assumptions

Below are some of the assumptions that have been considered for the Supermarket Web Service to work correctly:

- The Client Supermarket Program should be less than 100 meters from the Supermarket Server
- The Supermarket Server should be hosted on a PC with adequate resources. A server would have been preferable.
- Appropriate networking facilities from the WAP54G to the Supermarket Server.

Measuring Time Taken for Web Service Request

The time taken for the web service request is monitored using the timer implemented.

Scenario 1:

The Web Service request is first monitored over a wired network. Measurements are taken and the time taken for the Web Service Request is noted.

Scenario 2:

Then the system is deployed over a wireless infrastructure as explained in the "Implementation Section" and results for the time taken for the Wireless Web Service Request is noted down.

10 values for each web request both for the first and second scenario are taken and tabulated as shown below. Time is measured in millisecond

Table1: Time taken for Web Service Request

	Scenario 1 (Wired Network)	Scenario 2 (Wireless Network)
Time t1	7110	9652
Time t2	6825	9788
Time t3	6927	9503
Time t4	6936	9617
Time t5	6808	9607
Time t6	6831	14058 *
Time t7	6958	9825
Time t8	7004	9568
Time t9	6606	9874
Time t10	6800	9863
Average Time (T)	6881	9699

Discussions:

- The average time taken for a web service request over a wired network is 6881 millisecond whereas the average time taken for a web service request over a wireless infrastructure is 9699 millisecond. (It is to be noted that the time t6 for Scenario has been considered to be an out-of-range value and has not been considered when calculating the average time.
- The average time taken for a web service over a wireless infrastructure is more and this can be accounted by the fact that the wireless network had a lower bandwidth and a greater latency.

Measuring the availability of the system

The system should be able to provide service required a large percentage of the time. The system's availability was measured by doing 50 Web Service requests.

Scenario1: Measuring the number of successful web service requests over a wired network

Scenario 2: Measuring the number of successful web service requests over a wireless network.

These values have been tabulated as shown below.

Table2: Availability of System

	Scenario 1 (Wired Network)	Scenario 2 (Wireless Network)
Number of successful web service request out of 50	50	48
Availability	100%	96%

Discussions

There are 2 times when the wireless web service did not work and the error message is as shown below

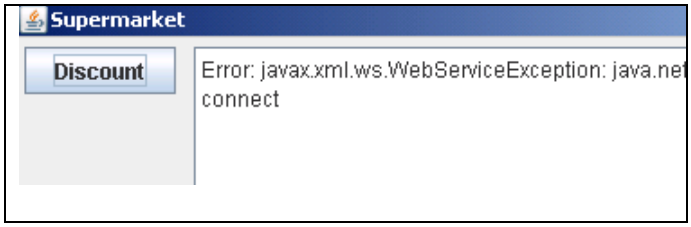

Figure8: Error Message

This is due to broken connection between the client and the server and this has led to failure of the web service requested.

- The availability when using the wired network is 100% whereas the availability for the wireless network is 96%.
- It would have been preferable to take measurements on different days so as to give a more adequate sampling.

Signal Strength of Wireless Infrastructure

During the testing phase, it has been observed that signal strength is an important factor to get a successful wireless web service request.

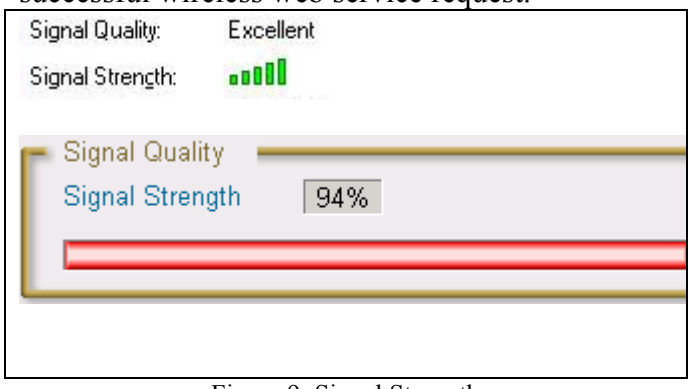

Figure 9: Signal Strength

The following are factors that have been observed to influence signal strength:

- Building materials such as steel and concrete walls can shorten the range of the signals.
- Physical layout of the area can interfere with the signals and cause them to be dropped.
- Electronic noise from cell phones and other electrical appliances can interfere with the transmission of the signals.

To provide adequate signal strength over long distances, repeaters can be used. The WAP54G has the option to behave as a repeater and another such device can be configured to act as a repeater.

Another means so as to avoid decrease in signal strength due to obstacles, is to fix the WAP54G either on the ceiling or on walls. This is shown below.

Figure 10: WAP54G wall mount

5 DISCUSSIONS AND CONCLUSION

General Discussions

It can be said that the performance of wireless web services is reasonable as compared to wired web services though the time taken for a wireless web service request is more. Taking into considerations the possible improvements, it is believed that this time taken can be further reduced**.** Hence it can be said that web services operate more or less suitably in a wireless environment. Furthermore it should be noted that the advantages of using such a wireless infrastructure are immense. This means that a customer can from anywhere and at anytime, request for useful information without having to go to a specific terminal or place.

Incremental Prototyping as a means for developing wireless web services is also believed to be appropriate because this ensures that a web service request is working fine over a wireless infrastructure and from there on, additional functionalities can be added incrementally. How the Web Service request would have reacted to the wireless infrastructure, was unsure in the beginning. It would have been pointless developing and implementing a whole complex system to find in the end that the system does not work over a wireless infrastructure.

It can also be said that the availability of the wireless web service is more than expected. From

the 50 samples of web service requests that have been taken, 48 have been successful wireless web service requests. This is also due to the fact that the devices/equipments are known to be among the best available on the market and are from manufacturers (D-Link and Linksys) that are pioneers in the wireless field. Another criteria that affects the availability of the wireless web service request for sure, is the distance of the client from the Wireless Access Point (WAP54G). The further it is from the WAP54G, the weaker is the signal strength and this can lead to unsuccessful wireless web service requests. Furthermore, obstacles might diminish the signal strength and one solution that has been mentioned to alleviate this problem of signal strength is to make use of repeaters.

The usability of the system, as it has been mentioned previously, is guided by two factors. The length of time it takes to request for a particular wireless web service and the ease with which users are able to use the system. As far as the time taken is considered, it is believed to be reasonable and suggestions have been made so as to further decrease this time taken. A set of users have been asked to test the interface and to give their feedback and suggested improvements. Most users found the interface very simple to use and believed that the information received were very helpful.

Conclusion

This project has focused on the development of a wireless web service for a supermarket and on the suitability of web services for implementing wireless application integration. It can be said that the web service operated more or less suitably in a wireless environment. Furthermore the availability and the usability of the system have been measured and this yielded good results. Incremental prototyping as well proved to be a beneficial technique.

This study also highlights which web server architecture is more appropriate and makes a comparative study between the traditional wired web service and wireless web services. The availability and usability of the proposed system were measured in view of investigating the suitability of the proposed system. The average time taken while using the developed application for a web service request over a wired network is 6881 millisecond whereas the average time taken for a web service request over a wireless infrastructure is 9699 millisecond. The latter is more and this can be accounted by the fact that the wireless network had a lower bandwidth and a greater latency. This time taken can be further reduced by taking into considerations the possible improvements suggested. The system's availability was also measured by comparing Web Service requests using the developed application both on a wired and on wireless infrastructure. The availability when using the wired network was 100% whereas the availability for the wireless network was 96%. It can be said that the performance of wireless web services is reasonable as compared to wired web services though the time taken for a wireless web service request is more. Hence it has been shown that web services operate more or less suitably in a wireless environment. To ensure a good quality of service and to provide adequate signal strength over long distances, repeaters can also be used.

What is interesting is that this provides new means and ways so as to provide customer satisfaction by providing the customer with useful information. Without any doubt, this will help him during his shopping and furthermore this information can be accessed, anytime and anywhere, within the supermarket without having to go to a specific place or terminal. Future works can ultimately include the use of a portable bar-code mechanism to retrieve the prices of goods that have been displayed in the supermarket.

6 REFERENCES

[1] Yuan, M.J., Long, Ju. Java Readies Itself for Wireless WebServices. http://www.javaworld.com/javaworld/jw-06-2002/jw-0621-wireless.html(2014)

[2] UDDI: Universal Description, Discovery, and Integration (UDDI), Technical report, UDDI.ORG. http://www.uddi.org (2014)

[3] Web Services Description Language (WSDL) 1.1. http://www.w3.org/TR/wsdl (2014).

[4] Limbu, D.K., Wah, L.E. and Yushi, C., Wireless Web Services Clients Development – Using Web Services Standards and J2ME Technology http://www.itsc.org.sg/pdf/Journal%202004/Section_Four_04/Four_Fire_3.pdf (2014)

[5] Snell, D. and Kulchenko, P., Programming Web Services with SOAP, First Edition (2001)

[6] Blake, M.B., Nowlan, M.F., Kona, S., Bansal, A. Annotating UDDI Registries for Composite Services, ACM Symposium on Applied Computing (SAC 2009), Honolulu, Hawaii, (2009)

[7] Benatallah, B., Dumas, M., Sheng, M., Ngu, A. Declarative Composition and Peer-to-Peer Provisioning of Web Services. IEEE International Conference on Data Engineering, IEEE Society, San Jose, California, (Feb 2002).

[8] Mitchell, S., Blake, M.B., Cunningham, D., and Gopalan, S. A SOA-driven Content Discovery and Retrieval Approach, IEEE International Conference on Electronic Commerce (CEC/EEE 2008), Washington, DC, (2008)

[9] Blake, M.B. and Nowlan, M.F. Taming Web Services from the Wild, IEEE Internet Computing Vol. 12, no. 5, 2008, pp. 62-69, (Sept/Oct 2008)

[10] Srirama, S.N., Jarke, M., Prinz, W., A Mediation Framework for Mobile Web Service Provisioning, edocw, 10th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW'06), (2006)

[11] Balani, N.,Deliver Web Services to mobile apps, IBM developerWorks, (2003)

[12] Steele, R., A Web Services-based System for Ad-hoc Mobile Application Integration, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.94.6517&rep=rep1&type=pdf (2014)

[13] Hongzhen, X., Hua, Q. and Maolan L, Nutrition Catering System of Hospitals Based on Web Services, IEEE International Conference on Integrating Wireless and Wired Network, Multimedia Information Networking and Security (MINES 09), (2009)

[14] Kim, K., Lee, H. and Chung, K., A Distributed Proxy Sever System for Wireless Mobile Web Service,15th IEEE International Conference on Information Networking, (2001)