

# Web Service Provisioning on Android Mobile Host

Kishor S. Wagh.

Research Scholar, SGG'S Nanded.  
Marathwada Mitra Mandal's Institute of Technology,  
Pune

R. C. Thool, Ph.D

Professor, Department of Information Technology  
SGGS College of Engineering and Technology,  
Nanded

## ABSTRACT

This paper discusses Web Service provisioning from Android Smart Phone. Web Service provisioning is the art of hosting and offering Web Services for Android based mobile device. This is due to advancements in mobile device manufacturing and wireless communication ability. This has changed the role of the mobile devices of Web service consumers to Web service providers. Today there is not a single real-time reasonable approach exists to host lightweight Web Services on Android smart phone. This paper describes the design and implementation details of the Android mobile host. Implementation of Android mobile host uses open source technologies such as Android, Java programming language, Android libraries, and SQLite database. The goal is to create an efficient and reliable Android mobile host as a web service provider that includes web services deployed on the mobile host and clients access the content. The main aim is to create lightweight weight framework for Android host that takes care of resource limitations and battery life. The manuscript presents a detailed performance evaluation of REST architecture and compares the results with SOAP standard.

## Keywords

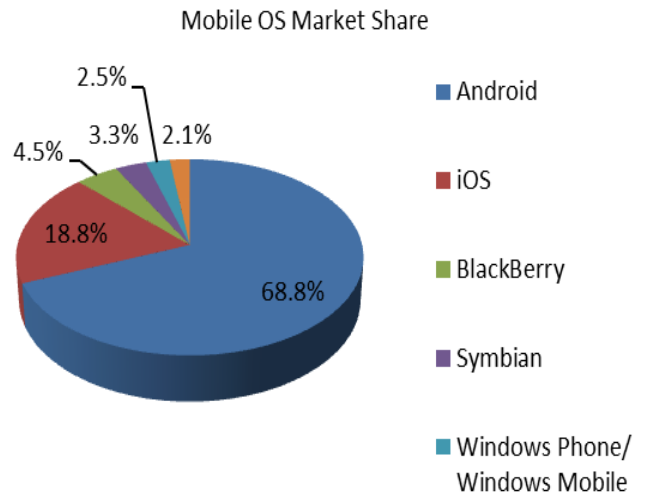
Android; Mobile web-server; Web Services; mobile Web Service provider; LTE.

## 1. INTRODUCTION

In recent years, the number of mobile devices with powerful processing and more memory addressing capacity has been increased. Day by day popularity of Android Smartphones are increasing and increasing the data transfer rate of wireless network. In future it is constantly going to increase more cable and more powerful with interesting features. Due to this, it is possible to develop and deploy new web services on smart phone. The number of smart phones worldwide counts about 1.8 billion units.

On the other hand, these huge numbers of smart phones represent a large number of heterogeneous devices with respect to the operating systems that the smart phones are currently using. According to [1], there were at least five different operating systems for smart phones available in the market, and their distribution shown in Figure 1.

Moreover 90% of the world population owns a cell phone and day by day transmission rates have increased by 3G, 4G and LTE (Long Term Evolution) as well as Wi-Fi enabling of these mobile devices to get access to Internet anywhere any time. Due to these improvements and the open source technologies for smart phones are enabling the mobile device work as a service provider.



**Fig 1: Distribution of different operating systems for Smartphones in 2012 (Source: IDC Worldwide Mobile Phone Tracker, April 12, 2013)**

According to [2] there are several reasons for hosting a web services on mobile phone to:

- Test locally client web applications without having to consume network access which might be limited by cost or availability in some areas.
- Allow third-party applications on other phones or other platforms to get access to your phone remotely. These needs strong security mechanisms that are provided in part by the network level authorizations.
- Run local Android applications that are leveraging the internal Web browser and behaving like regular hypermedia applications instead of using GUI's specific to Android.

The potential of mobile web service illustrated with a hospital search engine scenario. A web service is deployed on android smart phone that provides the clients to get access to an optimal search engine interface for searching hospitals, blood bank status in a particular city.

This paper describes a framework that provides web service from Android Mobile phone under real life settings and using now available open source technologies. Rest of paper organized as follows: II) Related work. III) Key challenges occur when web services provided by mobile phone. IV) Implementation of framework in detail with simulation results. V) Finally, last section concludes the paper and highlights future research directions.

## **2. RELATED WORK**

According to the Gartner research (June 15, 2001), “Web services are loosely coupled software components delivered over Internet standard technologies.” In short, Web services are self-describing and modular business applications that expose the business logic as services over the Internet through programmable interface and where IP used as finding ways to subscribe and invoke those services. A web service (WS) is software components that accessed over the Internet using standard protocols and well-defined interfaces. Web services are accessible using the Simple Object Access Protocol (SOAP) and the Hyper Text Transfer Protocol (HTTP).

From last decade, hosting web services on resource constrained mobile device has started. Hosting web service on a mobile device has two approaches: 1. Hosting entire web service on a mobile device, 2. Offloading the entire web service application to peer or remote node. Mobile devices used as a web service provider presented first by IBM [3]. This work presents shopper-kiosk specific scenario, where the web service hosted on a mobile device and addressed some of key design issues of hosting web service on a mobile device. The general concept of mobile web service provisioning introduced by [4] and has implemented a prototype for hosting web service on mobile host. Lightweight SOAP server architecture for mobile device proposed by [5]. The proposed server architecture is useful to give the access to web services via HTTP. The above discussed approach focuses on hosting web service on a mobile device. Some researchers [6] and [7] have explored the uses of mobile web services in peer-to-peer environment. In [8] author discussed middleware for Mobile web service provisioning. Mobile web service partitioning framework proposed by [9]. In this author investigated application partitioning algorithms for web service partitioning so that part of a web service executed on remote computing node to reduce the burden on a mobile host. In [10] discusses the web service distribution mechanism using fuzzy logic and describes the web service offloading and data migration schemes. In our paper [11], as per comparisons between SOAP and REST architecture, REST architecture is more suitable for a resource constraint Mobile Host. In continuation of research, this paper describes the design and implementation of a mobile host on Android smart phone for providing web services using REST architecture.

This paper explores the possibility to use the mobile phone such as Smart Phones as a Web Service provider. The paper presents a detailed architecture for Mobile Web Service Host, using HTTP GET/POST and HTTP Tunneling.

## **3. KEY CHALLENGES**

The designing of a mobile Host on android mobile device faces some key challenges. These key challenges are

- **Resources Limitation:** Processing of web service requires more CPU power, memory and battery but unfortunately the mobile devices limited in terms of each of these resources. This is a big challenge to researchers to consider resource imitations of mobile

devices and host web service on it without hampering it's basic functionality.

- **Hardware/Software Diversity:** Another key challenge is diversity in software and hardware used in mobile devices. The Mobile host implemented using open source technologies but different Operating System requires different implementation. So uploading web services on mobile devices is an expensive due to diversity of available mobile devices hardware, operating systems and even manufactures.
- **Light Weight Web Service Toolkits:** There is a number of the web service tool-kit and execution environments available for fixed servers and fixed networks but these tool-kits are not suitable for resource constrained mobile devices. These tool-kits are too heavy and it requires more resources. Mobile devices need light weight environment for hosting web services so that devices can have enough free resources to do the other core functionality such as making or receiving phone calls.
- **Weak Wireless Signals:** This is due to non-proper infrastructure provided by mobile operators. This inherits weak wireless signals that can cause connection drops and loss of packets. This also affects the battery consumption.
- **Data Transmission Speed:** This happens due to above reasons and mobile devices not providing more inbound and outbound for communication. Not proper 3G, LTE or 4G full flexed infrastructure available in India.

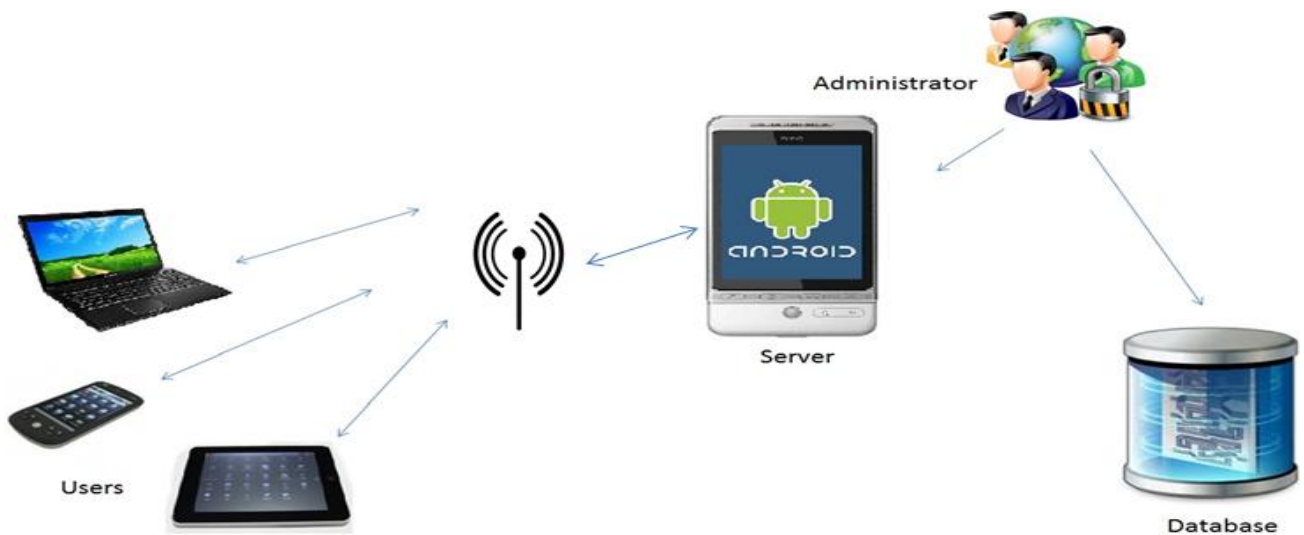
## **4. IMPLEMENTATION AND TEST RESULTS**

The major goal of this work is to study and analyze the behavior of a mobile host on the Android smart phone. The overall system architecture is shown in figure 2. The deployed web service of the Mobile Host is fully compatible with the usual Web interfaces so that clients will not notice any difference. The size of the Mobile host is a very small so that it requires very less processing power, memory and battery consumption.

This paper focuses on REST architecture. REST is the architecture of the web as it works by increasing the scalability and performance of the web application as compared to SOAP architecture.

The Characteristics of some methods used in REST architecture:

- **GET:** Safe, Idempotent, Cacheable
- **PUT:** Idempotent
- **DELETE:** Idempotent
- **HEAD:** Safe, Idempotent
- **POST:**



**Fig 2: System Architecture of Mobile Host Scenario**

Idempotent means request can do multiple times, Safe means no side effects using that method and Cacheable means method can able to pre-execute.

Some of the characteristics that should have while building Restful web service on mobile devices:

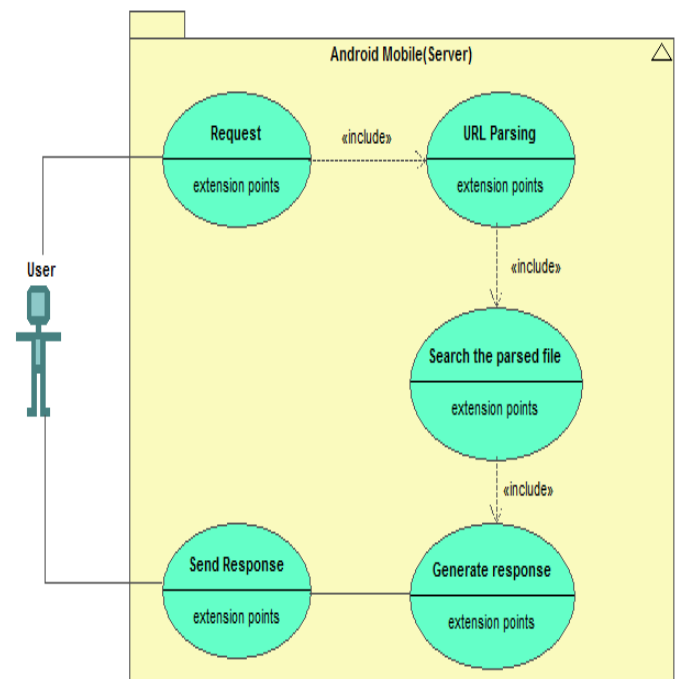
- Application state functionality extracted into resources, those resources are uniquely addressable using a universal syntax for using hypermedia links.
- All resources share a uniform interface for transferring the state between client and server consisting of a constraint set of well-defined operations, a constraint of content types, and optionally support code on demand.

The following sections give a brief description, how android mobile provides the web service to clients or consumers.

#### 4.1 Use-Case Analysis

In order to meet the goal is to carry out a mobile host on Android Smart Phone. The purpose of this diagram is to capture the dynamic aspect of a system. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system shown in figure 3. Here user can interact with a system from a web browser whether it is a mobile web browser or PC's web browser. User can just type the URL in a web browser, after that all other activities handled by a mobile host itself like URL Parsing activity which has sub activities like search parsed value in a directory and generate response depending upon search result and send response to the user. URL parser is our contribution to parse given URL and return the result. From understanding point of view user has any mobile device such as smart phone, tablet or laptop. In use case uses two actors to use functionality of the mobile Host / Server. Android Mobile Server software is a piece of software running on a mobile device that provides the web service to requested user. This mobile Host/Server is less in size as compared to available mobile servers in the market.

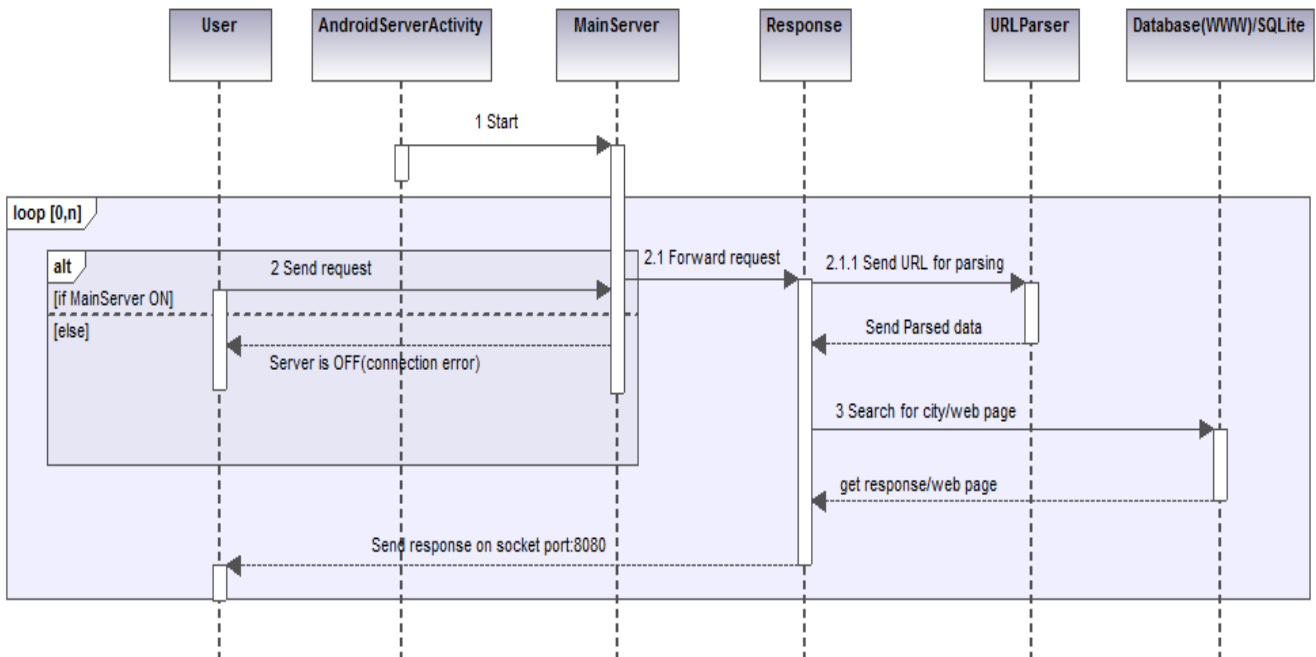
The user is another actor act as a web service client. The client uses the any web browser for performing requests to the web service running on Android Mobile Host/Server.



**Fig 3: Use case description of the developed mobile host on Android Mobile**

#### 4.2 Sequence And Activity Diagram

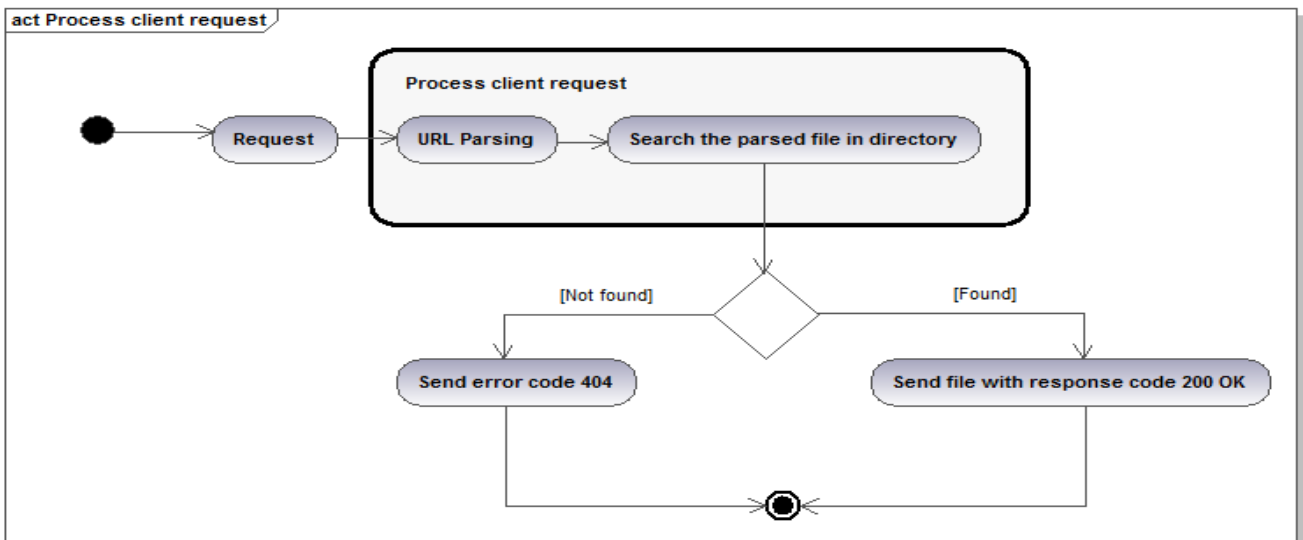
Sequence diagram gives the exact communication sequence between the web service client and Android Web Service Provider shown in figure 4. After the starting the web server, web server continuously is hearing the request from the client. When the client sends a request for web service, Android server passes client request to the URL parser through Response class. URL parser fetches data from SQLite database and send back to response class. Response class populates the data on the client.



**Fig 4: Sequence diagram for the communication between an Android web service provider and its client**

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Figure 5 represents the flow from one activity to another activity. The activity

diagram described as an operation of the system. Activity diagram describes how flow is moving from client to Mobile Web Service Provider.



**Fig 5: Activity diagram of client and a sever communication**

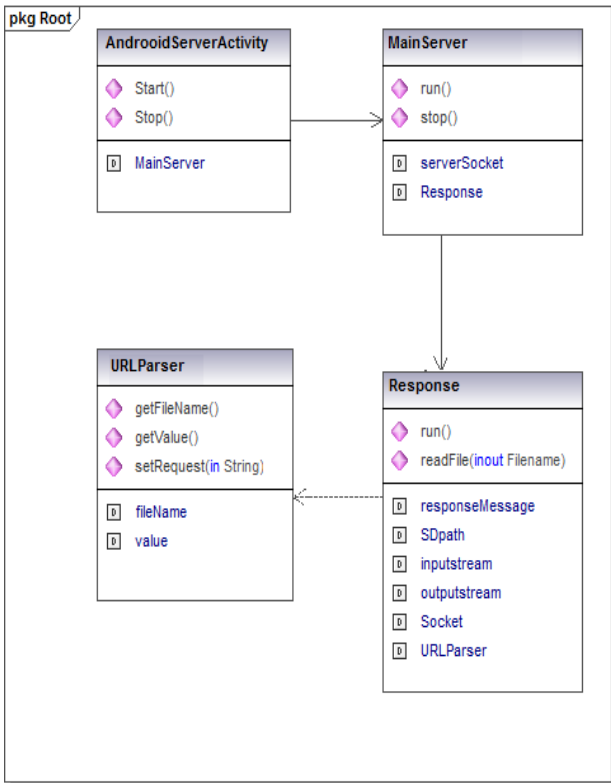
### 4.3 CLASS DIAGRAM

Class diagram provides an overview of the Mobile Host by describing the objects and classes in the server and the relationships between them. Figure 6 shows the class diagram of Android Mobile Host/Server. Start () used to activate Main Server thread and Stop () used to stop the thread. The main Server class creates a server socket to accept connection from client. Run () used to accept requests from client and send response by creating Respond class by creating a response class thread. Stop () used to stop Main Server Thread. Response class used to send a response to the client. This class fetches the URL and send to the parser. The URL parser parse the URL and sends return results to response class. Then

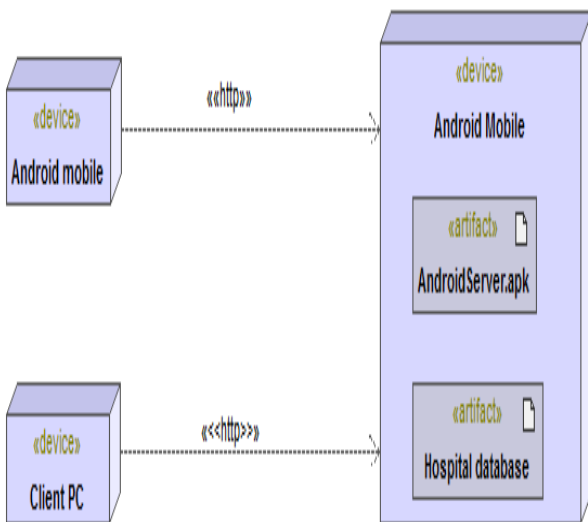
the response class object send response to the web service client.

### 4.4 Deployment Diagram

Deployment diagrams used as to visualize the topology of the physical components of a system. Figure 7 describes the hardware and software components of a mobile host. Here multiple clients send a request through HTTP to Android Mobile Host. AndroidServer.apk and test application data deployed on Android Mobile host.



**Fig 6: Class Diagram of Android Mobile Host**

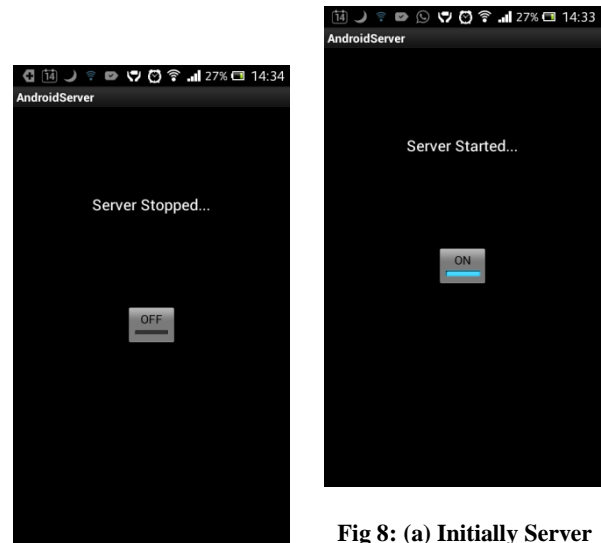


**Fig 7: Deployment Diagram of Mobile Host and Client**

### 4.5 Android Mobile Host Results

For the testing functionality of the mobile Host / Server use Hospital emergency web service that hosted on Android Smart Phone and that returns the name of Hospital with other important information requested by the client while accessing sever. Initially server is off so we have to switch it on to get it into listening mode shown in figure 8 (a). So after pressing the toggle button server switches to listening mode shown in figure 8 (b), i.e. wait for any request from a web service client to connect. Now Android server is on listening mode and waiting for clients' requests. When a request comes from a web service client to server, Server process sends a response

to the web service client. The most important is user can use his/her android smart phone as usual when server is running.



**Fig 8: (a) Initially Server OFF Fig 8: (b) Server is ON**

From web service client as a laptop is sending request to the Android Mobile Server shown in figure 9.



**Fig 9: A laptop client sending a request to the Android Mobile Server**

Response from Android Mobile Server to client is shown figure in 10.



**Fig 10: The response from Android Mobile Host to client**

The request processed by the server and then generates the response. Processing the request means server parse the requested URL and fetch the filename from it and search that file in the directory, and generate response code as per result i.e. 404 for file not found and 200 if the file is found.

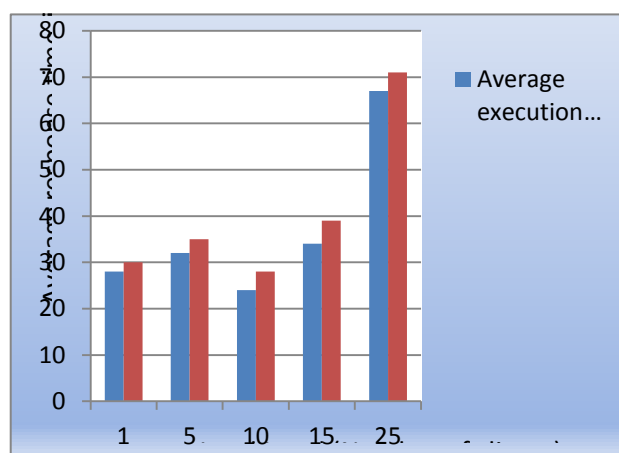
After finding a file from the directory it written to a socket which eventually seen on browsers. The first page has a text box and submit button. If a user writes “Pune” in the text box and press submit button it returns the list of hospitals with other details. While testing sending the number of requests at a time to mobile server and results noted and compared with SOAP architecture results shown in Table1.

**Table 1. Performance Analysis of REST Web Service**

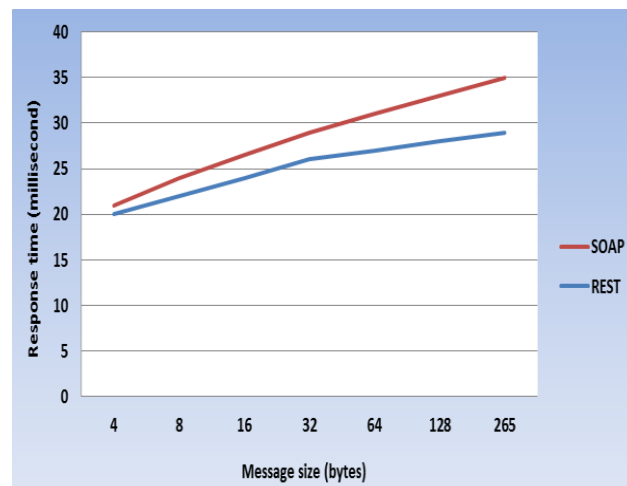
Load Level	Total No. of Reqt	Total No. of Resp	No. of Success Resp	% of Error Resp	Avg. Execu Time (MS)
1	57	57	57	0.0%	28
5	107	107	107	0.0%	32
10	122	122	122	0.0%	24
15	136	136	136	0.0%	34
25	143	143	143	0.0%	67

**Table 2. Performance Analysis of SOAP Web Service**

Load Level	Total No. of Reqt	Total No. of Resp	No. of Success Resp	% of Error Resp	Avg. Execu Time (MS)
1	57	57	57	0.0%	30
5	107	107	107	0.0%	35
10	122	122	122	0.0%	28
15	136	136	136	0.0%	39
25	143	143	143	0.0%	71



**Fig 11: Performance Analysis of REST VS SOAP**



**Fig 12: Time Required to Process Message in REST VS SOAP**

## 5. CONCLUSION

Today’s modern powerful Android mobile device is used as a web service provider. The Mobile Host is developed on a smart Android phone and that process service request and sends the response to the client. The clients access the Mobile Hosted web services via HTTP protocol. In Hospital search, blood bank, and temp converter scenario, Mobile Hosts that can themselves offer services in a true mobile client-server environment. A Web Service provider shows feasibility of this approach in terms of resource consumption, standard compliance, and performance. After testing Mobile Host for various web services scenarios it paves scope of web services on Mobile phones in the client-server and distributed mobile information networks.

Future plan to carry out a hybrid framework for web service provider using REST architecture and that can support concurrent access by HTTP and SMS so that it decrease battery consumption.

## 6. REFERENCES

- [1] IDC Worldwide Quarterly Mobile Phone Tracker from <http://www.idc.com/getdoc.jsp?containerId=prUS239460> 13, April 12, 2013.
- [2] <http://www.wiki.restlet.org>
- [3] S. McFaddin, C. Narayanaswami and M. Raghunath, “Web Services on Mobile Devices – Implementation and Experience”, In the 230 Proceedings of the Fifth IEEE Workshop on Mobile Computing Systems & Applications (WMCSA’03), pp. 100-109, Monterey, CA, U.S.A., October 2003.
- [4] S. Srirama, M. Jarke and W. Prinz, “Mobile Web Service Provisioning”, In the Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT/ICIW 2006), pp. 120-128, Guadeloupe, French Caribbean, February 2006.
- [5] L. Pham and G. Gehlen, “Realization and Performance Analysis of a SOAP Server for Mobile Devices,” In the Proceedings of the Eleventh European Wireless Conference, Vol. 2, pp. 20-27, Nicosia, Cyprus, April 2005.

- [6] G. Gehlen and L. Pham, “Mobile Web Services for Peer-to-Peer Applications”, In the Proceedings of the Second IEEE Consumer Communications and Networking Conference (CCNC’05), pp. 427-433, Las Vegas, NV, U.S.A., January 2005.
- [7] M. Hassan, “Mobile Web Service Provisioning in Peer to Peer Environments” 2009 IEEE International Conference on Service Oriented Computing and Applications, Page1-4.
- [8] F. Aijaz, S. Adeli and B. Walke, “Middleware for Communication and Deployment of Time Independent Mobile Web Services”, In the Proceedings of the IEEE International Conference on Web Services (ICWS’08), pp. 797-800, Hawaii, HI, U.S.A., July 2008.
- [9] M. Asif, S. Majumdar and R. Dragnea, “Partitioning the WS Execution Environment for Hosting Mobile Web Services”, In the Proceedings of the 2008 IEEE International Conference on Services Computing (SCC’08), pp. 315-322, Honolulu, HI, U.S.A., July 2008.
- [10] AlShahwan, F., Moessner, K. And Carrez, F.; “Providing and Evaluating the Mobile Web Service Distribution Mechanisms Using Fuzzy Logic,” Journal of Software, ISSN 1796-217X vol. 7, no. 7, 2012.
- [11] Wagh K., Thool R., “A Comparative Study of SOAP Vs REST Web Services Provisioning Techniques for Mobile Host”, Journal of Information Engg and Applications, Vol. 2, No. 5, 2012, pp.12-16.
- [12] Reto Meier, “Professional Android Application Development”, Wiley Indian Edition, 2011.
- [13] Hirsch, F.; Kemp, J.; Ilkka, J., “Mobile Web Services Architecture and Implementation”, John Wiley & Sons, Ltd., England, 2006.
- [14] Syed A. Ahson., Mohammad Ilyas., “Mobile Web 2.0: Developing and Delivering Services to Mobile Devices” CRC Press, ISBN 978-1-4398-0082-9.