CLOUDHEALTH: DEVELOPING A RELIABLE CLOUD PLATFORM FOR HEALTHCARE APPLICATIONS

1 ROHIT GUPTA
M-Tech Student
Department of computer science & Engg
Buit Bhopal

2 MOHIL DWIVEDI
M-Tech Student
Department of computer science & Engg
Buit Bhopal

3 DR ANJU SINGH
Assistant Professor
Department of computer science & Engg
Buit Bhopal

4 DR DIVAKAR SINGH
Head of the departament
Department of computer science & Engg
Buit Bhopal

ABSTRACT

With the evolution of fog computing, process takes place domestically during a virtual platform instead of in a centralized cloud server. Fog computing combined with cloud computing is a lot of economical as fog computing alone doesn't serve the aim. Inefficient resource management and cargo equalization results in degradation in quality of service in addition as energy losses. Traffic overhead is redoubled as a result of all the requests are sent to the most server inflicting delays that cannot be tolerated in attention eventualities. to beat this downside, the authors are consolidating fog computing resources in order that requests are handled by foglets and solely crucial requests are sent to the cloud for process. Servers are placed domestically in every town to handle the close requests so as to utilize the resources with efficiency at the side of load equalization among all the servers, those results in reduced latency and traffic overhead with the improved quality of service.

Keywords: Fog computing, Utilization, Load equalization, Cloud computing

1 INTRODUCTION

Cloud computing is associate degree rising computing technology that uses the net to take care of giant applications and knowledge servers to supply services to purchasers or finish users. It is that the delivery of computing services over the internet through computing resources. a lot of exactly, cloud computing is that the virtualization and central management of data-centered resources. It refers to remotely managing workloads over the web during a knowledge center. Cloud computing eliminates the price of hardware, software, and controls in addition as manages the on-the-spot knowledge centers. Knowledge will be reflected at multiple surplus sites on the cloud provider's network. It generates knowledge backup, provides a disaster recovery, and ensures business continuity with easier and less-expensive services. Cloud suppliers supply computing services to the purchasers and charge them supported the usage of those services. Thanks to inherent downside, some applications cannot work with efficiency on the cloud. Downside of low information measure arises as knowledge can't be trans- mitted to the cloud with the frequency because it is being generated. For example, in crucial things like coronary failure once giant amounts of information is to be sent back to edge nodes when process at the cloud, delay cannot be tolerated. Hence, to beat this issue, the construct of fog computing was introduced.

The term fog computing (also called fogging), introduced by CISCO, is associate degree extension of cloud computing. it's wont to ease wireless knowledge transfer to distributed devices within the IoT network paradigm. Fog computing brings computing resources and application services nearer to the sting wherever knowledge is being generated. The quantity of information sent to the cloud for process and analysis is reduced to an oversized extent by victimization fog computing. It reduces the traffic on cloud (i.e., main server) and is ready to maintain load between resources, that improves quality of service (QoS) and security.

Fog computing creates a virtual platform providing net- operating, computation, and storage services on the brink of the amount wherever knowledge is being generated, i.e., within the middle of cloud knowledge centers and finish users.

Foglet resource management is a crucial issue of fog computing. The load among foglets isn't balanced (some foglets is also full whereas others could stay idle). Inefficient resource management and cargo equalization results in degradation in QoS and energy management of foglets. FOCAN (Fog Computing design Network) [1] may be a multi-tier architecture projected to beat these problems. It ensures low energy usage, response time reduction, and reduced network overhead. Abdel et al. in [2] has illustrated the role of IoT in education domain. at the side of this, the authors have provided definitions, basic ideas, characteristics, and challenges of IoT. Its role in creating economical and effective selections is additionally demon- started during this work.

This paper integrates fog computing thus that:

- ❖ Most of the requests are handled by cloudlets and solely crucial requests are sent to the cloud for any process.
- ❖ native servers are placed in every town to handle the close requests in order that resources are utilized
- ❖ with efficiency and cargo is balanced among all the servers
- ❖ at the side of that, latency and traffic overhead is reduced with the improved QoS.

The rest of the document is organized as follows: Section a pair of contains temporary description of connected work. In Section three, analysis parameters are outlined. Section four describes the projected design. Section five explains the operating of one foglet. Finally, the document is over in Section half dozen.

2 LITERATURE REVIEW

This section describes existing work related to differentaspects of fog computing. Fog computing is a widely spread domainand many researchersare exploring it nowadays. We categorize these contributions according to the issues faced in foglet resource management. The authors in [1], have proposed FOCAN (multi-tier structure) for energy consumption, resource utilization, and latency. Low energy usage, response time reduction, and reduced network overhead are achieved while load balancing, security, and privacy are not considered. In [3], the authors proposed definitions for mobile edge computing and motivation of mobile edge computing by considering different applications. This is done to bring cloud ser- vices, resources, and mobile edge computing. The authors resolved the issue regarding WAN latency for delay sensitive applications in accessing the cloud resources; how- ever, the problem of traffic overhead isnot addressed in this paper.

The problem of load balancing in cloud computing is discussed in [4]. Load balancing algorithms and techniques are used to serve the purpose. Response time is decreased with VM. In [5], a previous allocation state of VM is not saved and the algorithm is executed each time when new request for VM allocation is made. Hence, round robin with server affinity is pro- posed. With the proposed algorithm, the authors are able to reduce the response time and processing time. The authors in [6] address the problem of response time and processing time. A combination of two algorithms is used: ESCE and throttled algorithm. A comparison of ESCE, throttled algorithm, and round robin is done. From the results, it is clear that ESCE and throttled algorithms performed well, then round robin and time and cost are reduced by using these two algorithms.

The goal of [7] is to locate user demand and to minimize the tremendous growth of mobile traffic. Higher bandwidth cost and energy consumption are considered. The author uses a three-tier mobile fog cloud architecture to achieve the goal. By using this architecture, fog computing absorbs intensive mobile traffic and relieves good data transmission. Data storage, security and privacy issues, and costly and energy-hungry data are the application areas discussed in [8]. Maintaining and operating sensors directly from cloud servers are non-trivial tasks. Multi-layer telemedicine architecture is proposed using Intel Edison, Raspberry pi, case studies on various types of physiological data and body sensor networks. Reliability is achieved with reduced traffic overhead, and the overall performance is enhanced. Data trimming and resource utilization are the application areas discussed in [9]. The authors presented smart gateway focused communication. A single-hop gateway and multi-hop gateway are used. Efficient resource utilization for better data trimming and pre-processing, and reduced traffic head are achieved. In [10], the authors have proposed the hierarchical model and application architecture, information system infrastructure framework, and resource manage- ment mechanism for data backup, data management, and system monitoring. Burden is reduced on network traffic, and system efficiency is also improved while security is not considered.

In paper [11], the authors have considered the security issues faced in fog computing paradigm. The authors have studied a real typical attack (man in the middle attack), analyzing memory consumption and CPU. Data security is more as compared to other SOTA. Real-time schedul- ing algorithm is proposed in [12], to recover connection failure and to retain services for vehicles that lose the fog server. Control overhead and failure recovery time was decreased by 55%.

The paper [13] deals with three challenges. (1) Design of information sensing nodes in body sensor networks. (2) Collection, storage, and analysis of large amounts of heterogeneous data. (3) Energy efficiency of edge devices. The authors propose a service-oriented fog computing architecture to overcome these challenges of data reduc- tion, low power consumption, and high efficiency. Dubey et al. built a prototype using Intel's Edison in order to show the efficiency of the proposed architecture. Achievements of the authors include reduction in storage and power along with the reduced logistics requirements. The main focus of the authors in [14] is to enhance the IoT-based health monitoring systems used for diversified environments. The authors proposed an IoT-enabled healthcare system architecture to demonstrate the efficiency of band-width utilization, emergency notification, and quality of service assurance. To determine the efficiency of fog computing in healthcare applications, it is implemented on a case study of ECG. This paper implemented different fog computing services like location awareness, interoperability, graphical user interface with access management, distributed database, and real-time notification mechanism.

The authors in [15] are dealing with the problem of resource management. A methodology for management and estimation of resources is proposed which is known as relinquish probability. How many resources are going to be used and whether all the requested resources are consumed or not, this cannot be predicted. With the model proposed by Aazam et al., one can determine the right amount of resources required which results in reduction of resource wastage and profit-cut for CPS and fog. Authors in [16] proposed the new fall detection algorithm as early algorithms for fall detection have too much false alarm rate and missing rate. However, with the proposed algorithm, high specificity and high sensitivity is achieved with a minimum response time and energy consumption. The paper [17] presents the fog computing architecture for emergency alerts. A lot of work has already been done on emergency management despite that the architecture proposed in this paper is simple and efficient. By a single button click, users can send alert and then the application decides on its own which department should be informed [18]. Moreover, it automatically informs the patients family by sending them a message. Overall delay with fog computing is reduced six times than other cloud cases. In paper [19], iFogSim is introduced, modelled, and simulated in IoT, edge, and fog environment. In particular, the authors described two case studies and demonstrated effectiveness of iFogSim.

3 EVALUATION PARAMETERS

This section gives a brief description of the different evaluation parameters along with their importance in the healthcare domain.

3.1 Load balancing

It refers to the distribution of application or network traffic among different servers in order to enhance the capacity and reliability of the applications. It is the distribution of the task performed by a single computer into multiple computers so that more work gets done at the same time. By this distribution of workloads and the computing resources, we can manage the workload demands in a better way by allocating resources (requests in this case) among multiple servers and will serve users rapidly [28]. This will lead to a high availability and increased performance rate. The aim of this research is to efficiently utilize load which can be done by distributing workload among different servers. If one server is too busy in handling requests from the clients and other servers are idle or have a less amount of requests, then it will transfer some of the load on the nearby server which has no or less requests. Hence, by this distribution of workload, we can achieve efficient utilization of energy and resource consumption.

3.2 Latency/response time

Latency refers to delays which usually occur when any component of the system waits for another component to complete the task. Basically, it is the time taken by the processor to handle the request, i.e., from the moment of transmission till the time it is received back to the client after all the processing [29]. In case of any delay in this processing time, it is considered as latency. It is among one of the drawbacks of the cloud computing which is handled by fog computing. In the scenario discussed in this paper, delays can not be tolerated as in life-threatening situations delays may cause any mishap.

3.3 Quality of service

It is the ability to provide better services to network traffic over different technologies. The goal of QoS is to allocate the lead containing dedicated bandwidth, managed and controlled latency, and jitter. QoS is not a one time deployment in a varying network environment, rather it is an ongoing and fundamental part of a network design.

3.4 Bandwidth

It is the volume of information that can be transmitted per unit of time. The maximum amount of data that can be transmitted over a specific network or internet in a given amount of time that the transmission medium can handle is termed as bandwidth. It only describes the speed of the network and does not tell how fast data is moving from one location to another. The amount of band- width required depends on what you are planning to do with your internet connection. The higher the bandwidth, more data are transmitted, and hence, in a particular time, more processing of data can be done due to the large amount of data transmission. One can also limit the band- width for a certain task. This control of bandwidth is set

Table1 State-of-the-ArtWorktocomparecontributionsbydifferentaspects

Parameters						
Reference papers						
	Traffic overhead	Quality of service	Load balancing	Latency/response Time	Energy consumption	Bandwidth
[1]			×			
[3]	×		×		×	
[4]	×	×			×	×
[5]	×	×	×		×	×
[6]	×	×	×		×	×
[7]		×	×	×		
[8]	×	×	×		×	
[9]		×		×	×	×
[10]			×	×	×	×
[11]	×	×	×	×	×	×
[12]	×	×	×		×	
[13]	×	×	×	×		
[14]	×		×		×	
[15]	×	×	×		×	×
[16]	×	×	×			×
[17]	×	×	×		×	×
[33]	×	×	×		×	

3.5 Traffic overhead

It refers to the amount of extra resources that do not have any direct relationship with the production. The amount of processing time required by the system including the operating system, the utility which supports the application programs, and the installation of any of the particular feature will add to the proportion already needed by the program. It is also defined by Martin et al. in [30] as the processing time of a processor in which it is engaged in the transmission or reception of each message; during this period of time, the dedicated processor cannot perform any other operations.

3.6 Energy consumption

This is the amount of energy used in a particular process or system, or energy consumed by an organization or a society. It is the total amount of energy required to pro- vide the services to the end users. Energy conservation will lead to more efficient systems thus reducing the wastage of resources.

3.7 Security

It refers to securing the data, most probably sensitive data, of the end users or clients of the system. The protection of the data to make sure that only authorized personnel have access over the data or services provided by a certain system and to ensure the safety against attacks. It is one of the biggest challenges which fog computing is facing, and in order to overcome this access, control should be applied.

3.8 Privacy

Utilization of data while protecting the privacy of the individuals. It refers to the anonymity of an individual and determines whether the data or information of any individual or organization can be accessed by third parties or not, and when, how, and to whom this information is to be revealed.

3.9 Support of mobility

To facilitate traffic forwarding from one node to another which results in change of location of the information. It is useful to handle and allocate resources efficiently among fog nodes [31]. In this work, it is referred as the ability to transfer some load from an over burdened fog node to the idle one. The mobility has an important influence on communication as well [1].

3.10 Interoperability

It refers to the ability of system or software to use the information even after exchange of the data or informa- tion. This use of the same tool or software on a variety of platforms is considered as interoperability. In this scenario, if we move or exchange data from one foglet to another, then it would be usable and useful for transmit- ting the same information.

3.11 Data storage

Recording of information in a storage medium for future use is termed as data storage. With the increasing demand, data storage and data processing in the IoT have become an issue. To resolve this problem, utilization of cloud com- putting was introduced which was later replaced by fog computing.

3.12 Network management

It refers to monitoring and managing a wide range of computer networks which might be a burden for fog com- puting unless some of the techniques are applied on it. Applying these techniques on fog computing may be a challenging task and may lead to mismatch with the goals of efficiency and latency.

3.13 Resource management

It is the efficient and effective management of all the avail- able resources in a best possible way. It is responsible for allocation of resources and maintenance of resource pool in a distributed fashion.

3.14 Jitter

It is the delay between the received packets. It is considered as the variation in the data flow between two systems which might occur because of network congestion. Jitter can be reduced with fog computing.

4 PROPOSED ARCHITECTURE

The architecture proposed in this paper contains two servers, namely, cloud server and fog server. Patients will send the request to the fog server via IoT. The IoT is the network of objects and the connectivity among these objects which allow them to connect and exchange data. Figure 2 depicts that the requests from patients of different cities are handled by different servers. All the requests from the cluster of Location Aare received by foglet A, foglet B is handling the requests from Location B cluster, similarly foglet C is dealing with the requests coming from the cluster ofLocation C, and requests from Location D cluster are handled by foglet D. In return of the request received, an acknowledgement will be sent to the patients that the request is received. On the same fog layer, status of the patient will be checked and if the condition of patient is critical and after making the copy of that request, it will be forwarded to the cloud server without any delay. For example, the heart ECG of the patient is determined if the condition of the patient is critical, then it will be sent to the cloud for further processing; otherwise, the fog server will handle this request. If any of the server is overloaded, then the requests will be shifted to another nearby server

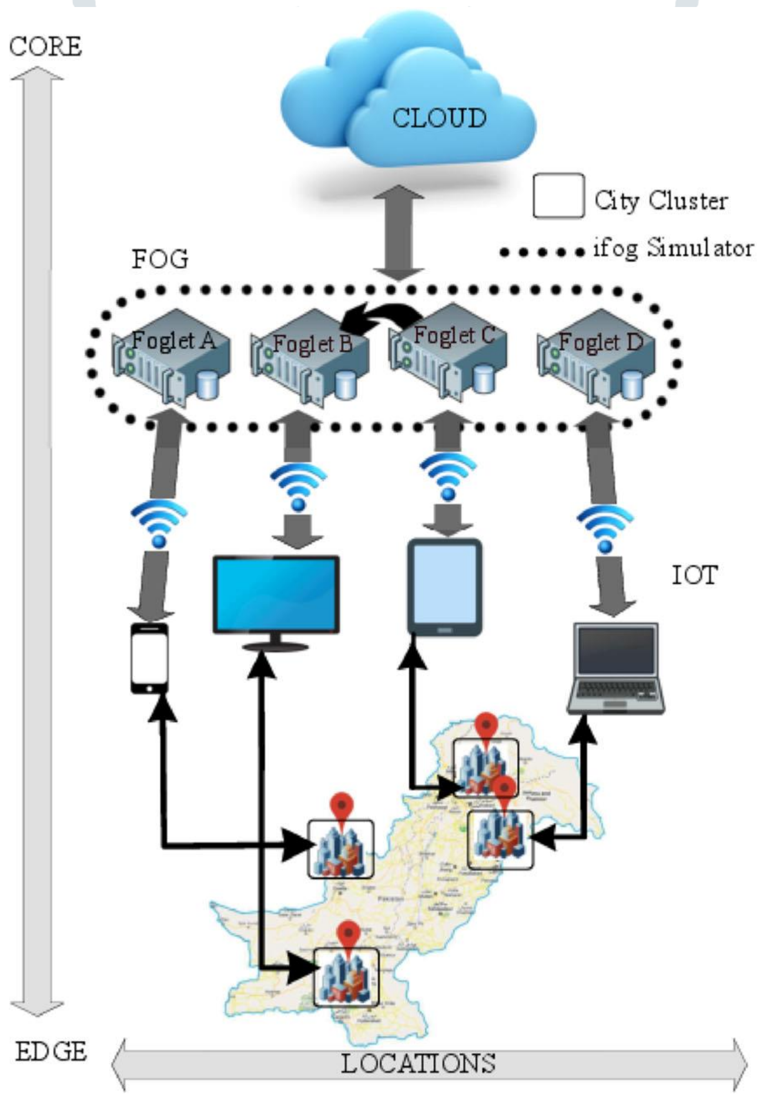

Fig. 2 Proposed architecture

which will respond to the request according to the condition of the patient. Under normal circumstances, the request will be handled by the fog servers; however, in case of a critical condition of the patient, the request will be forwarded to the cloud server for processing. Fog layer which is the middle-ware is maintaining the copy of the request for future references and in order to maintain the record of that patient. Cloud server will generate alerts, and according to the condition of the patient, it will send some precautionary measures to the patient. Fog layer is also handling the load balancing. Capacity of each foglet is determined according to the capacity of each server requests handled. If the number of requests on a server exceeds the level, then it will be forwarded to some other nearby server and hence balancing of incoming requests will be done.

5 Methods/experimental setup for foglets

In this section of the paper, step-wise flow of activities is given. Working of a single component or a foglet is described in detail in Fig. 3.

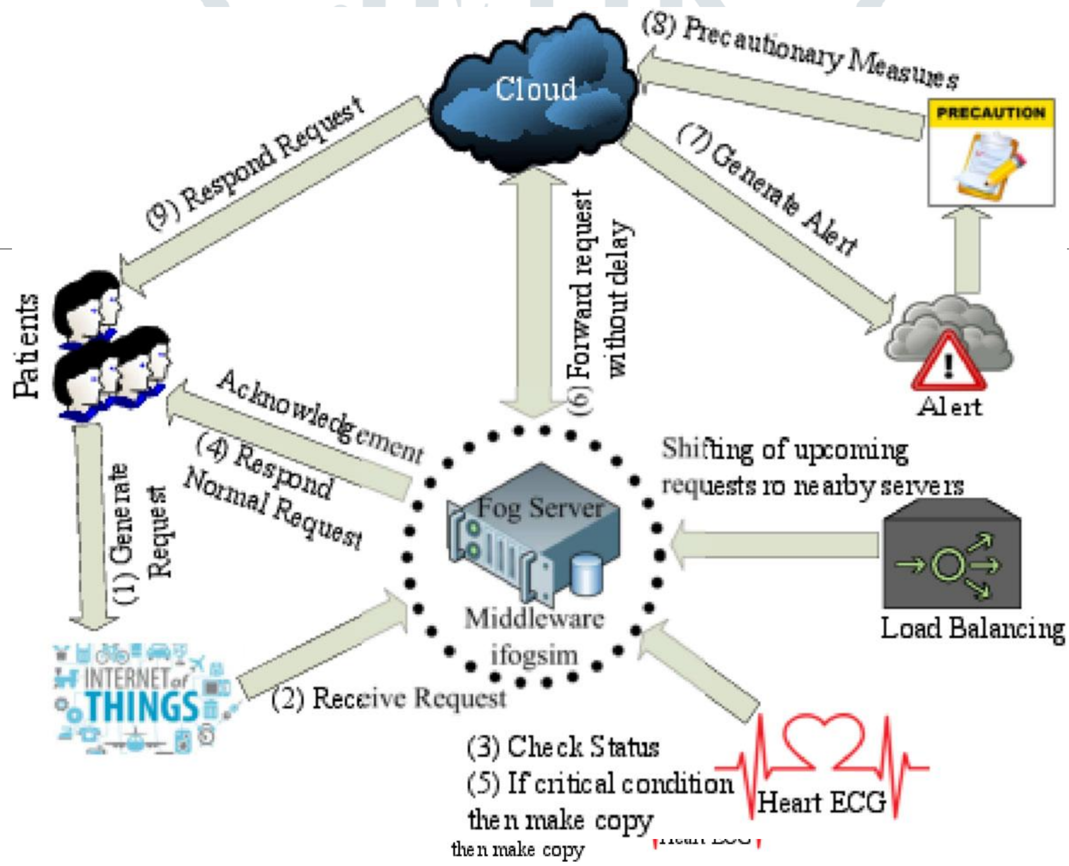

Fig.3 Flowofactivities

For experimental validation, we have used the iFogSim framework, which is an industry standard for the development of fog models along with IoT integration. Moreover,it provides enhanced capabilities to measure the impact of resource management techniques through important parameters such as latency, energy consumption, and cost.

5.1 Generate request

In the first step of the proposed architecture, patients are going to send the requests to the fog server through IoTs which include objects connected to the internet. Both normal and critical patients can generate request and send that to the server for processing.

5.2 Receive request

The fog server will receive the request from the patient and an acknowledgement will be sent from the fog server to the patient that the request is received and it is being processed.

5.3 Check status

The status of the patient will be checked in this step. If the condition of the patient is normal, then the fog server will handle this request in order to avoid the traffic on the cloud server. Otherwise, the request will be sent to the cloud.

5.3.1 Respond normal request

After checking the status of the request, its response will be sent back to the patient. The fog server will respond the request of the normal patient. A normal heart rate of an adult and aged people ranges from 60 to 100 beats per minute. However, the normal heart rate for children is between 70 and 100 beats per minute. These ranges may vary according to the health condition of each person. Hence, all the requests which lie between these ranges are considered normal and these requests will be handled by the fog server.

5.3.2 Critical patient

If the condition of the patient is critical, then the fog server will make a copy of the request received from patient and after that it will forward the request to the cloud without any delay.

5.4 Response time

The request from the patient who is in critical condition is forwarded to the cloud server without any delay because in safety critical systems or in life-threatening conditions delays are not affordable. A request is sent to the cloud server because most of the requests are handled by the fog server so that there would be less traffic on the cloud. Hence, within the minimum possible time, a request can be sent to the cloud to avoid any mishap.

5.5 Alert

After receiving the request, the system will generate an alert considering the condition of the critical patients.

5.6 Precautions

Precautionary measures regarding the condition of the critical patient are then generated and sent to the patient that he needs to follow immediately. For instance, if the heart beat of a person exceeds the normal range, then the person will be notified through an alert and precautionary measures will be suggested to them which are expected to be followed by the patient.

5.7 Respond request

Request could be responded for both stable and unstable patients.

5.7.1 Stable condition

After precautionary measures, if the patient gets stable, then they will be informed that their condition is normal now and that they can continue with their normal routine.

5.7.2 Unstable condition

In case the condition of the patient does not get stable, then they will be suggested to rush towards the hospital immediately in order to avoid any mishap.

6 CONCLUSION

Fog computing being the extension of cloud computing provides ease in wireless data transfer to the distributed devices in the IoT paradigm. It serves as a middle-ware by bringing computing resources and application services closer to the edge where data is being generated. Previously, with the cloud computing, all the data was being

sent to the cloud for processing which results in delays and latency was not considered. However, with the fog computing, only a limited amount of information will be transmitted to the cloud; hence, bandwidth will be enhanced to a large extent with the reduced latency and delay along with packet loss. The aim of this research is efficient utilization of resources. By placing fog servers between cloud and end users, delays can be reduced. As in healthcare scenarios, delays cannot be tolerated. We are distributing the load evenly on all the servers so that the overburdened servers may not lead to breakdown and all servers are having approximately equal amounts of load. This is done by shifting the load from the overburdened server to the nearby server which is idle or which has less load. Even distribution of load on the fog layer will lead to load balancing and efficient utilization of resources. In this work, we examined how utility is affected by various parameters. More specifically, we have examined fog server utilization. In order to evaluate utilization of fog nodes, we contribute to the capabilities of the iFogSim tool.

REFERENCES

1. P. Naranjo, G. Vinueza, Z. Pooranian, M. Shojafar, M. Conti, R. Buyya, FOCAN: A Fog-supported Smart City Network Architecture for Management of Applications in the Internet of Everything Environments. arXiv preprint arXiv. 1710, 01801 (2017)

2. M. Abdel-Basset, M. Mohamed, G. Manogaran, E. Rushdy, Internet of things in smart education environment: Supportive framework in the decision-making process. Concurrency Comput. Pract. Experience, e4515

3. E. Ahmed, M. H. Rehmani, Mobile edge computing: opportunities, solutions, and challenges. Futur. Gener. Comput. Syst. 70, 59–63 (2017)

4. R. Soumya, A. D. Sarkar, Execution analysis of load balancing algorithms in cloud computing environment. Int. J. Cloud Comput. Serv. Archit. (IJCCSA). 2(5), 1–13 (2012)

5. M. Komal, A. Makroo, D. Dahiya, Round robin with server affinity: a VM load balancing algorithm for cloud based infrastructure. J. Inf. Process. Syst. 9(3), 379–394 (2013)

6. N. Shaveta, G. Raj, Comparative analysis of load balancing algorithms in cloud computing. Int. J. Adv. Res. Comput. Eng. Technol. 1(3), 120–124(2012)

7. T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. Wei, L. Sun, Fog computing: Focusing on mobile users at the edge. arXiv preprint arXiv. 1502, 01815 (2015)

8. D. Harishchandra, A. Monteiro, N. Constant, M. Abtahi, D. Borthakur, L.Mahler, Y. Sun, Q. Yang, U. Akbar, K. Mankodiya, in Handbook of

Large-Scale Distributed Computing in Smart Healthcare. Fog Computing inMedical Internet-of-Things: Architecture, Implementation, andApplications (Springer, Cham, 2017), pp. 281–321

9. M. Aazam, E. Huh, in 2014 International Conference on Future Internet of Things and Cloud, Barcelona. Fog Computing and Smart Gateway Based Communication for Cloud of Things, (2014), pp. 464–470. https://doi.org/10.1109/FiCloud.2014.83

10. H. Xingye, L. Xinming, L. Yinpeng, in 2010 International Conference on Computational and Information Sciences, Chengdu. Research on Resource Management for Cloud Computing Based Information System, (2010),pp. 491–494. https://doi.org/10.1109/ICCIS.2010.127

11. I. Stojmenovic, S. Wen, in 2014 Federated Conference on Computer Science and Information Systems, Warsaw. The Fog computing paradigm:Scenarios and security issues, (2014), pp. 1–8. https://doi.org/10.15439/2014F503

12. S. Park, Y. Yoo, Network Intelligence Based on Network State Information for Connected Vehicles Utilizing Fog Computing. Mob. Inf. Syst. 2017,9 (2017). https://doi.org/10.1155/2017/7479267

13. S. Yi, Z. Hao, Z. Qin, Q. Li, in 2015 Third IEEE Workshop on Hot Topics in WebSystems and Technologies (HotWeb), Washington, DC. Fog Computing: Platform and Applications, (2015), pp. 73–78. https://doi.org/10.1109/ HotWeb.2015.22

14. H. Dubey, J. Yang, N. Constant, A. M. Amiri, Q. Yang, K. Makodiya, inProceedings of the ASE BigData & SocialInformatics 2015 Article No. 14Kaohsiung, Taiwan - October 07 - 09, 2015. Fog data: Enhancing telehealth big data through fog computing, (2015)

15. T. N. Gia, M. Jiang, A. Rahmani, T. Westerlund, P. Liljeberg, H. Tenhunen, in2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing,Liverpool. Fog Computing in Healthcare Internet of Things: A Case Study on ECG Feature Extraction, (2015), pp. 356–363. https://doi.org/10.1109/ CIT/IUCC/DASC/PICOM.2015.51

16. M. Aazam, E. Huh, in 2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops), St. Louis, MO. Dynamic resource provisioning through Fog micro datacenter,(2015), pp. 105–110. https://doi.org/10.1109/PERCOMW.2015.7134002

17. C. Yu, S. Chen, P. Hou, D. Brown, in Networking, Architecture and Storage (NAS), 2015 IEEE International Conference on. FAST: A fog computing assisted distributed analytics system to monitor fall for stroke mitigation (IEEE, 2015), pp. 2–11

18. S. Yingjuan, G. Ding, H. Wang, H. E. Roman, S. Lu, in Future Information andCommunication Technologies for Ubiquitous HealthCare (Ubi-HealthTech),2015 2nd International Symposium on. The fog computing service for healthcare (IEEE, 2015), pp. 1–5

19. G. Harshit, A. V. Dastjerdi, S. K. Ghosh, R. Buyya, iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments. Softw. Pract. Experience. 47(9), 1275–1296 (2017)

20. M. Abdel-Basset, G. Manogaran, M. Mohamed, Internet of Things (IoT) and its impact on supply chain: A framework for building smart, secure and efficient systems. Futur. Gener. Comput. Syst. 86, 614–628 (2018)