International Journal of Multidisciplinary Research and Publications
ISSN (Online): 2581-6187

Client/Server Remote Control Administration System: Design and Implementation

Amira B. Sallow[1], Hivi Ismat Dino[2], Zainab Salih Ageed[3], Mayyadah R. Mahmood[4], Maiwan B. Abdulrazaq[5]
[1,3]Computer Science Department, Nawroz University, Duhok, Kurdistan Region – Iraq
[2, 4, 5]Computer Science Department, College of Science, University of Zakho, Duhok, Kurdistan Region – Iraq
Email address: amira.bibo @ nawroz.edu.krd[1], hivi.dino @ uoz.edu.krd[2], zainab.ageed @ nawroz.edu.krd[3], mayyadah.mahmood @ uoz.edu.krd[4], maiwan.abdulrazzaq @ uoz.edu.krd[5]

Abstract— *Networks of computers are everywhere, and it implemented in all sectors, including an IT, industrial, managerial sector, and need to be accessed remotely. Remote administration offers access to a remote device. As more departments of Information Technology centralize and reorganize to keep costs down, many remote sites are left with no IT support on-site. Remote computer administration is increasingly common due to the major cost advantages; many activities can be completed, and not needs to be personally accessed by the administrator. In this paper, the remote-control administration system has designed and implemented based on client/ server architecture. The main goal of the proposed system allows clients to request help from remote servers over the Network.*

Keywords— *Remote Control System, Remote Client, Remote Server, Monitoring, Controlling, Remote system Administration.*

I. INTRODUCTION

One of the most common techniques used on networked computers is their ability to operate on remote networks. That is, on a remote machine, the user needs to invoke an operation. [1] [2].

There are many cases in which network administrators face the problem of managing and controlling their separate computer networks when they are away from their offices [3]–[7]. In these cases, network administrators are dependent on third party reports to know their network status. Some even have to direct such third parties on how to resolve network issues on their behalf when they (network administrators) are out stationed, most of which lead to jamming of networks and other related problems [8].[9].

Remote access is the connection from a secondary location to a device other than that of the system's primary location being accessed [10]–[14]. This allows users to access a remote computer as if they sat directly behind that computer. This remote access by users to computers is possible because of remote access tools and by extension remote access protocols [15]–[20]. Such remote access protocols are tools used to develop applications via third parties [21].

In this paper, the remote-control administration system has designed and implemented. The proposed system allows users to request assistance from a remote server over the network, which includes a mechanism by system administrators that can directly connect to a remote computer and address users' specific issues. The remote server performs remote operations to control and monitor the remote client. The system has programmed with the C# programming language.

This paper is organized as follows. The next section 2, reviewed the related work. Section 3, "protocols" describes the network protocols. Section 4, "the client/server computing model and the internet" explains the client/server computing model and the internet. Section 5, "sockets and client/server communication" illustrates the socket and client/server computing. Section 6, "net remoting" explains net remoting. Section 7, "organization of the proposed system" describes in detail the structure of the suggested system. Section 8, "remote execution "shows the steps of implementing the remote administration. Finally, conclusions are presented in section 9.

II. RELATED WORK

Jang [22] in 2012, presented a remote administration system for the Windows platform. The remote administration system was designed for client and server architecture in the Windows operating system. A client operation has the opportunity to allow the server access to the network. This feature is responsible for monitoring the packet for requesting on-network access to the client. The capability of the server system can connect to the client system. The connection is made by guiding the IP address of the client which we would like to access the server system. Visual Basic was used to design this device.

Manikandasamy [23] in 2013, aimed at controlling the PC through remote Java-enabled mobile phone access to the desktop. Corporate users can share resources and services without having to invest a large amount of capital on hardware and software infrastructure. The mobile device acts as a remote monitor collecting user data and showing the notification updates provided by the remote server. Operations such as internet access, email sending, and even device controls such as shutdown, log-off, and restart can access from the mobile phone.

Ganaa [8] in 2015 proposed innovative technology to provide massive support to network administrators by introducing a stable remote system administration software running on android smartphones to help them remotely control their servers while they (network administrators) are stationed using their smartphones. The Java eclipse used to develop Android app establishes a secure connection to a remote server

Amira B. Sallow, Hivi Ismat Dino, Zainab Salih Ageed, Mayyadah R. Mahmood, and Maiwan B. Abdulrazaq, "Client/Server Remote Control Administration System: Design and Implementation," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 3, Issue 2, pp. 5-11, 2020.

running a PHP application. The program was designed using the Remote Frame Buffer (RFB) protocol. Hence, this research integrated a self-signed Secure Socket Layer (SSL) certificate in the android app to allow safe encrypted connections between the android app and the remote server to be created. The proposed and implemented stable RFB protocol in the android app was compared with other current remote device administration applications such as Remote Desktop (RDP), and ICMP ping-command RFB protocols.

Ernest [8] in 2015, studied how Network Administrators use remote access devices as network administrators in their day-to-day activities. It also investigated how efficient and reliable these tools were in terms of the ability of the tools to return desired results to users.

Shaha [24] in 2016, introduced a remote lab monitoring program to correct the situation faced by the administrator/instructor to control the activities of the client over a LAN and to manage the computer network or LAN by implementing such software to perform operations capable of monitoring the entire computer lab by sitting on one chair by observing the activities of the client, sending messages to the client system, Once the user logs into the program, the attendance is automatically stored, and the students' screen linked to the teachers' computer will become shown.

Gupta [25] in 2018, implemented remote desktop control and monitoring framework to solve the problem faced by each company to look at work performed on their respective computers by staff/students. The program enables the monitoring and tracking of every action taken on a device by anyone. In addition to monitoring, the administrator can also manage the desktop of their client by getting access to manage their hardware on the keyboard, mouse, etc. to help them solve any problem or problem.

Magaña [26] in 2019 compared five of the most common remote desktop protocols and provided models based on self-similar processes for their traffic arrival. They had implemented as DaaS service in a public virtual cloud. The protocols were: PCoIP as used in the Amazon WorkSpaces, Microsoft RDP, TeamViewer, VNC (RFB), and Citrix ICA. The required network resources had assessed using a traffic model based on self-similar processes. Also enhance the quality of interaction experienced by the user, providing Mean Opinion Score (MOS) values in terms of image quality and interactivity. The results suggest that to assess the bandwidth specifications, the type of application running on the remote servers and the mix of users have to be considered.

III. PROTOCOLS

The Internet Protocol Suite is the collection of protocols that implement the [27], [28] protocol stack on which the Internet is working. After two of the many protocols that make up the suite, it is often called the TCP / IP protocol suite: The Transmission Control Protocol (TCP) and the Internet Protocol (IP) which were described as the first two. RFC 1122 is the authoritative guide on this subject [29]–[34].

The Internet model has developed as a solution to a practical issue of engineering. On the other hand, the OSI model was a more theoretical approach, which was developed at an earlier point of network evolution [35] [36]. The OSI configuration is also easier to understand but the TCP/IP model is the one in actual use. Before learning TCP / IP, it is useful to understand the OSI model, since the same principles apply, but are easier to understand in the OSI model. The "Fig.1" below. Attempts to show where different TCP / IP and other protocols in the original OSI model would reside [37]:

7	Application	HTTP, SMTP, SNMP, Telnet,...
6	Presentation	XDR, ASN.1, SMB,...
5	Session	ISO 8327 / CCITT X.225, RPC, NetBIOS,...
4	Transport	TCP, UDP, RTP, SPX,...
3	Network	IP, ICMP, IGMP, X.25, CLNP, ARP,...
2	Data Link	Ethernet, Token ring, PPP, HDLC,...
1	Physical	Electricity, Radio, Laser

Fig. 1. OSI layer

Generally, in the TCP / IP suite, the three top layers of the OSI model (Application, Presentation, and Session) are reduced to a single application layer [38]–[42]. Since the TCP / IP suite does not have a unified session layer on which higher layers are built, individual applications typically perform (or ignore) those functions. A simplified stack description for TCP/IP is shown below in "Fig. 2" [43]

7	Application	HTTP, FTP, Telnet,...
4	Transport	TCP, UDP ,...
3	Network	IP, ICMP, IGMP, X.25, CLNP, ARP,...
2	Data Link	Device Driver ,...

Fig. 2. TCP/IP Layers

A. TCP

TCP is a connection-based protocol which provides a reliable
data flow over the network between two connected devices.
TCP provides a point-to-point channel for applications in which reliable communication is needed [44]–[48]. The Telnet, File Transfer Protocol (FTP), and Hypertext Transfer Protocol (HTTP) are examples of services that required a reliable communication channel. The successful performance of these applications is the order in which the data is transmitted and received over the network [49].

B. UDP

UDP (User Datagram Protocol) is a protocol that sends independent data called datagrams, with no guarantees of arrival from one device to another. UDP is not like TCP-based

Amira B. Sallow, Hivi Ismat Dino, Zainab Salih Ageed, Mayyadah R. Mahmood, and Maiwan B. Abdulrazaq, "Client/Server Remote Control Administration System: Design and Implementation," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 3, Issue 2, pp. 5-11, 2020.

communication [50]–[54]. The UDP protocol provides for non-guaranteed communication between two network applications [55].

IV. The Client/Server Computing Model and the Internet

The Internet offers a variety of services that contribute to its attractiveness. Such resources include e-mail, newsgroups, transfer of files, remote access, and the Web Internet services have organized according to the architecture of the client/server [56]. User programs such as web browsers and file transfer programs build connections to the server, such as web servers and FTP servers. The clients make server requests, and the server responds to the requests by delivering the service the client needs [57].

A good example of client/server computing has provided through the Web. The clients are web browsers, and the servers are web servers. Browsers request HTML files from Web servers on the user's behalf by establishing a Web server connection and submitting requests for files to the server. The server receives requests for the files, retrieves the files and sends them over the defined link to the browser. The files are received by the browser and shown to the user browser window [58].

V. Sockets and Client/Server Communication

Clients and servers establish connections and connect using sockets. Connections are links of communication which are established using TCP over the Internet. There are also several client/server applications designed around the connectionless UDP. These applications often communicate through sockets [59]–[63].

Sockets are the endpoints of communication over the Internet. Clients build client sockets and connect the sockets to the server. The sockets are associated with a host address with a port address. The host address is the host's IP address on which the client or server software is located. The port address is the port of communication used by the client or server application. Server programs use the well-known port number associated with their application protocol [64]

A client communicates with a server by making a connection to the server socket. Afterward the client and server exchange information over the connection. Connection-oriented communication is more efficient than connectionless communication as the underlying TCP offers facilities for message-acknowledgment, error detection, and error-recovery. By using a connectionless protocol, the client and server connect to each other's socket by sending datagrams. The UDP is used for connectionless protocols. It does not support safe communication such as TCP [1], [65]

VI. NET Remoting

Remoting is the. NET equivalent of Java remote method invocation (RMI) and the Distributed Common Object Model (DCOM) of Visual Basic [66]–[70]. It makes it easier to use complex objects on remote computers, using the same syntax as if they were in the same app. The benefit that Remoting affords is the network infrastructure abstraction. [71] [72]

This performs simple implementing client/server applications in which the server is required to perform a variety of tasks based on client instructions [73] [74].

Users still need to create a client and server when using remote services. Users must also build an object that will perform whatever functions user needs. Both ends of the connection need to know the type of the object. The client will need to know the server's IP address and port. Other than that, .NET does the rest [75] [76].

VII. Organization of The Proposed System

The proposed system is controlled by two sides (Remote Server and Remote Client).

1) *Remote server:* consists of supervision and controlling applications as shown in "Fig.3", Server main interface is used to start the server to enter the client computer by choosing the computer name and the IP address from the clients' database.

Fig. 3. Server main interface

Also, add new clients to monitor by entering the computer name and IP address of the client that wanted to be monitored in the network as shown in "Fig.4a". Delete and update the client's computer name and IP address as in "Fig.4b".

a: Add new Client

Amira B. Sallow, Hivi Ismat Dino, Zainab Salih Ageed, Mayyadah R. Mahmood, and Maiwan B. Abdulrazaq, "Client/Server Remote Control Administration System: Design and Implementation," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 3, Issue 2, pp. 5-11, 2020.

b: Delete/Update Windows

Fig. 4. Remote server database operation Interfaces

2) *Remote client:* contained the shutdown, and change the password as in "Fig.5". After login, the client can perform any task on the computer.

Fig. 5. Remote Client Main interface

The client is able to shut down the client and prevent the server from controlling and close the monitor window in the server side as in "Fig. 6".

Fig. 6. Remote Client shutting Down interface

Also, the client can change the password that used to shut down the client from monitoring and controlling in the server side as in "Fig.7".

Fig. 7. Remote Client changing Password interface

VIII. REMOTE EXECUTION

The general representation of Remote execution and its mechanism are shown in the flowchart shown in "Fig.8".

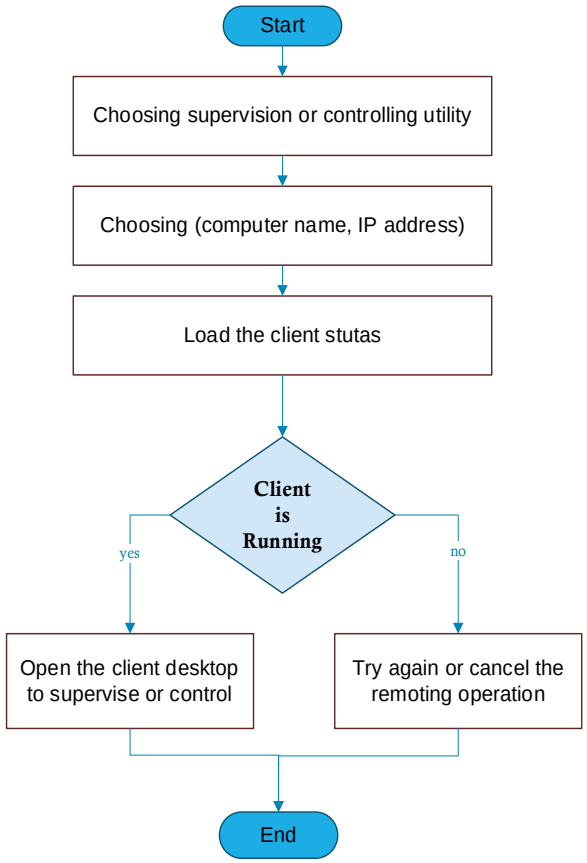


Fig. 8. Flowchart

When the remote server login to the client computer first monitoring style should be chosen, supervision or control, then select the client name from the given database list the storied all the clients' names and IP addresses as in "Fig.9".

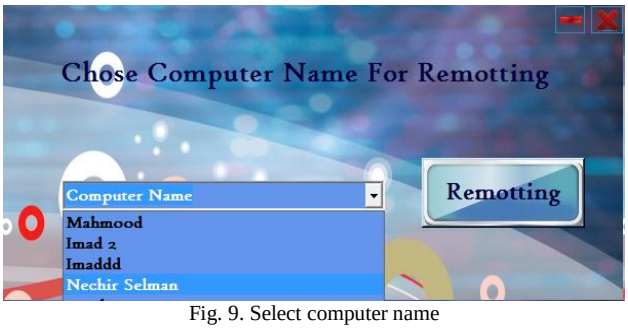


Fig. 9. Select computer name

After the remote server chooses the computer name, the client window will be loaded as in "Fig.10", this task takes a little period to check if the client is shut down or if there are any network problems.

Amira B. Sallow, Hivi Ismat Dino, Zainab Salih Ageed, Mayyadah R. Mahmood, and Maiwan B. Abdulrazaq, "Client/Server Remote Control Administration System: Design and Implementation," *International Journal of Multidisciplinary Research and Publications (IJMRAP)*, Volume 3, Issue 2, pp. 5-11, 2020.

Fig. 10. Loading interface

After checking if there is any problem the error windows will be appeared for the remote server asking to try again or cancel the remoting operation as in "Fig.11".

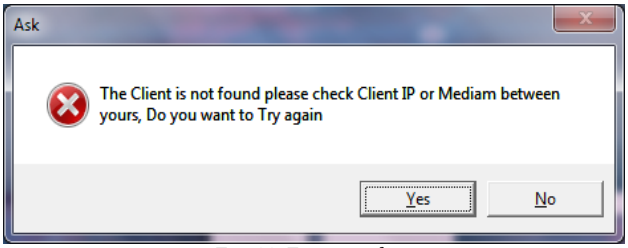
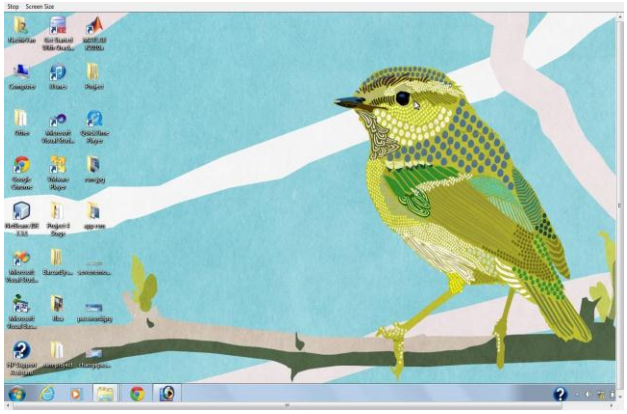
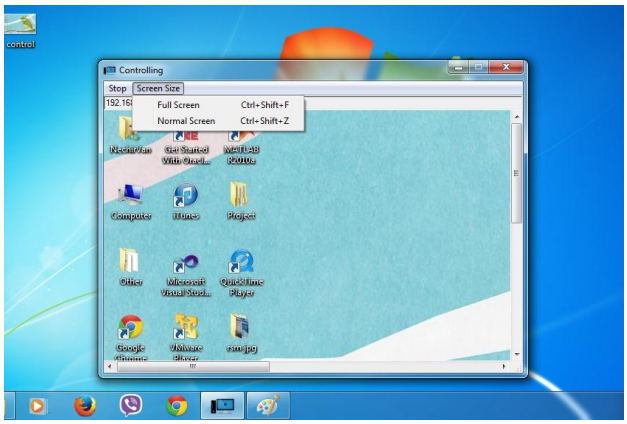


Fig. 11. Error interface

If the remote server login to the remote client computer successfully the environment will operate like that the remote server computer will be directly connected to the remote client system. The remote server can monitor the client system in full or normal screen size mode as in "Fig.12 a,b".



a: Control in full-screen Size



b: Control in Normal screen Size

Fig. 12. Screen control mode

IX. Conclusion

In this paper, the remote-control administration system has been designed and implemented deployed based on client-server architecture using the C# programming language. The following conclusions can be marked:

1. Virtual Remote Desktop Control System offers a process by which system administrators can directly connect to a remote computer and address clients' specific issues.
2. Remote Desktop Control System will help in controlling and supervision clients' desktop screens.
3. If the client accepts the invitation, the server's administrator can set up a remote session to monitor the user screen and communicate with the client; the mouse and keyboard could be monitored as an alternative.

References

[1] Lydia Parziale *et al.*, "TCP/IP Tutorial and Technical Overview," p. 1004.

[2] H. K. Shaikha and A. B. Sallow, "Mobile Cloud Computing: A Review," *Acad. J. Nawroz Univ.*, vol. 6, no. 3, p. 6, 2017, doi: 10.25007/ajnu.v6n3a96.

[3] S. R. Zeebaree, R. R. Zebari, and K. Jacksi, "Performance analysis of IIS10.0 and Apache2 Cluster-based Web Servers under SYN DDoS Attack," *TEST Eng. Manag.*, vol. 83, no. March-April 2020, pp. 5854–5863, 2020.

[4] S. R. Zeebaree, R. R. Zebari, K. Jacksi, and D. A. Hasan, "Security Approaches For Integrated Enterprise Systems Performance: A Review," *Int. J. Sci. Technol. Res.*, vol. 8, no. 12, Dec. 2019.

[5] S. R. M. Zeebaree, H. M. Shukur, L. M. Haji, R. R. Zebari, K. Jacksi, and S. M.Abas, "Characteristics and Analysis of Hadoop Distributed Systems," *Technol. Rep. Kansai Univ.*, vol. 62, no. 4, pp. 1555–1564, Apr. 2020.

[6] S. M. Mohammed S. R. M. Zeebaree, B. w. salim, R. R. Zebari, H. M. Shukur, A. S. Abdulraheem, A. I. Abdulla., "Enterprise Resource Planning Systems and Challenges," *Technol. Rep. Kansai Univ.*, vol. 62, no. 4, pp. 1885–1894, Apr. 2020.

[7] S. R. Zeebaree, K. Jacksi, and R. R. Zebari, "Impact analysis of SYN flood DDoS attack on HAProxy and NLB cluster-based web servers," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 19, no. 1, pp. 510–517, 2020.

[8] E. D. Ganaa, F. Twum, and J. B. Hayfron- Acquah, "A Comparative Study Of Remote Access Technologies and Implementation of a Smartphone App for Remote System Administration Based on a Proposed Secure RFB Protocol," *Int. J. Sci. Eng. Appl.*, vol. 4, no. 7, pp. 163–168, Jul. 2015, doi: 10.7753/IJSEA0404.1002.

[9] D. Abdullah and A. Sallow, "Developing Fault Tolerance Integrity Protocol for Distributed Real Time Systems," *AL-Rafidain J. Comput. Sci. Math.*, vol. 10, no. 1, pp. 187–193, 2013, doi: 10.33899/csmj.2013.163435.

[10] S. R. M. Zeebaree, L. M. Haji, I. Rashid, R. R. Zebari, O. M Ahmed, K. Jacksi, H. M. Shukur, "Multicomputer Multicore System Influence on Maximum Multi-Processes Execution Time," *TEST Eng. Manag.*, vol. 83, no. May/June, pp. 14921–14931, May 2020.

[11] R. Zebari, S. Zeebaree, K. Jacksi, and H. Shukur, "E-Business Requirements for Flexibility and Implementation Enterprise System: A Review," *Int. J. Sci. Technol. Res.*, vol. 8, pp. 655–660, Nov. 2019.

[12] R. R. Zebari, S. R. Zeebaree, and K. Jacksi, "Impact Analysis of HTTP and SYN Flood DDoS Attacks on Apache 2 and IIS 10.0 Web Servers," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 2018, pp. 156–161.

[13] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, "A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction," *J. Appl. Sci. Technol. Trends*, vol. 1, no. 2, Art. no. 2, May 2020, doi: 10.38094/jastt1224.

[14] O. H. Jader, S. R. Zeebaree, and R. R. Zebari, "A State Of Art Survey For Web Server Performance Measurement And Load Balancing Mechanisms," *Int. J. Sci. Technol. Res.*, vol. 8, no. 12, pp. 535–543, Dec. 2019.

[15] L. Haji, R. R. Zebari, S. R. M. Zeebaree, W. M. Abduallah, H. M. Shukur, and O. Ahmed, "GPUs Impact on Parallel Shared Memory Systems Performance," *Int. J. Psychosoc. Rehabil.*, vol. 24, no. 08, pp. 8030–8038, 21, May, doi: 10.37200/IJPR/V2418/PR280814.
[16] O. Alzakholi, L. Haji, H. Shukur, R. Zebari, S. Abas, and M. Sadeeq, "Comparison Among Cloud Technologies and Cloud Performance," *J. Appl. Sci. Technol. Trends*, vol. 1, no. 2, Art. no. 2, Apr. 2020, doi: 10.38094/jastt1219.
[17] H. Shukur, S. Zeebaree, R. Zebari, O. Ahmed, L. Haji, and D. Abdulqader, "Cache Coherence Protocols in Distributed Systems," *J. Appl. Sci. Technol. Trends*, vol. 1, no. 3, pp. 92–97, 2020.
[18] H. Shukur, S. Zeebaree, R. Zebari, D. Zeebaree, O. Ahmed, and A. Salih, "Cloud Computing Virtualization of Resources Allocation for Distributed Systems," *J. Appl. Sci. Technol. Trends*, vol. 1, no. 3, pp. 98–105, 2020.
[19] L. M. Haji, S. R. Zeebaree, O. M. Ahmed, A. B. Sallow, K. Jacksi, and R. R. Zeabri, "Dynamic Resource Allocation for Distributed Systems and Cloud Computing," *TEST Eng. Manag.*, vol. 83, no. May/June 2020, pp. 22417–22426, 2020.
[20] H. I. Dino, S. R. M. Zeebaree, A. A. Salih, R. R. Zebari, Z. S. Ageed, H. M. Shukur, L. M. Haji, S. S. Hasan, "Impact of Process Execution and Physical Memory-Spaces on OS Performance," *Technol. Rep. Kansai Univ.*, vol. 62, no. 5, pp. 2391–2401, June. 2020.
[21] Aditya Bomble, Anish Ghag, Nilesh Hase, Umesh Jagtap, "Remote Administration and Network Security Tool," *Int. J. Sci. Res. Dev.*, vol. 3, no. 01, p. 3, 2015.
[22] S.-J. Jang, "Implementation of the Remote Control and Management System in the Windows O.S," p. 5, 2012.
[23] R.Manikandasamy, "Remote Desktop Connection Using Mobile Phone," *Int. J. Sci. Eng. Technol. Res. IJSETR*, vol. 2, no. 8, 2013.
[24] Vipul Shaha, Amit Arabhavi, Chetan Barage, Suraj Chavan, Yogesh Karande, Bhagyashri Kelkar, "Remote Lab Monitoring," *Int. Res. J. Eng. Technol.*, vol. 3, no. 3, 2016.
[25] A. K. Gupta, S. Goswami, A. Nasa, and D. Gupta, "Remote Desktop Monitoring and Controlling," vol. 05, no. 03, p. 2, 2018.
[26] E. Magaña, I. Sesma, D. Morató, and M. Izal, "Remote access protocols for Desktop-as-a-Service solutions," *PLOS ONE*, vol. 14, no. 1, p. e0207512, Jan. 2019, doi: 10.1371/journal.pone.0207512.
[27] F. A. Zulkifle, R. Hassan, R. M. Othman, And A. B. Sallow, "Supervised Classification And Improved Filtering Method For Shoreline DetectioN," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 20, pp. 5628–5636, 2017.
[28] A. B. Sallow, A. A. H. Alkurdi, and Z. A. Sulaiman, "Proposed System for Educational Augmented Reality Smart Book," *Acad. J. Nawroz Univ.*, vol. 8, no. 3, p. 61, Aug. 2019, doi: 10.25007/ajnu.v8n3a397.
[29] D. B. Abdullah and A. B. Sallow, "Developing Fault Tolerance Integrity Protocol for Distributed Real Time Systems," *AL-Rafidain J. Comput. Sci. Math.*, vol. 10, no. 1, pp. 187–193, 2013.
[30] Z. N. Rashid, S. R. Zebari, K. H. Sharif, and K. Jacksi, "Distributed Cloud Computing and Distributed Parallel Computing: A Review," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 2018, pp. 167–172.
[31] L. M. Haji, S. R. Zeebaree, K. Jacksi, and D. Q. Zeebaree, "A State of Art Survey for OS Performance Improvement," *Sci. J. Univ. Zakho*, vol. 6, no. 3, pp. 118–123, 2018.
[32] D. Q. Zeebaree, H. Haron, A. M. Abdulazeez, and S. R. Zeebaree, "Combination of K-means clustering with Genetic Algorithm: A review," *Int. J. Appl. Eng. Res.*, vol. 12, no. 24, pp. 14238–14245, 2017.
[33] Z. Subhi RM and J. Karwan, "Effects of Processes Forcing on CPU and Total Execution-Time Using Multiprocessor Shared Memory System," *Int. J. Comput. Eng. Res. TRENDS*, vol. 2, no. 4, Art. no. 4, 2015.
[34] R. K. Ibrahim, S. R. M. Zeebaree, and K. F. S. Jacksi, "Survey on Semantic Similarity Based on Document Clustering," *Adv. Sci. Technol. Eng. Syst. J.*, vol. 4, no. 5, pp. 115–122, 2019, doi: 10.25046/aj040515.
[35] A. B. Sallow and S. R. Hussain, "Multi-Agent System for Supporting and Managing Real Estate Marketing," *Acad. J. Nawroz Univ.*, vol. 9, no. 3, pp. 54–62, 2020.
[36] A. B. Sallow and H. Kh. Shaikha, "Optical Disc and Blood Vessel Segmentation in Retinal Fundus Images," *Acad. J. Nawroz Univ. AJNU*, vol. 8, no. 3, pp. 67–75, 2019.
[37] S. R. Zeebaree, A. B. Sallow, Bzar Khidir Hussan, and Sundos Mohammad, "Design and Simulation of High-Speed Parallel/Sequential Simplified DES Code Breaking Based on FPGA," presented at the 2019 International Conference on Advanced Science and Engineering (ICOASE), Zakho - Duhok, Iraq, Iraq, Apr. 2019, doi: 10.1109/ICOASE.2019.8723792.
[38] A.-Z. S. R. Zeebaree, A. Z. Adel, K. Jacksi, and A. Selamat, "Designing an ontology of E-learning system for duhok polytechnic university using protégé OWL tool," *J Adv Res Dyn Control Syst Vol*, vol. 11, pp. 24–37.
[39] A. AL-Zebari, S. R. Zeebaree, K. Jacksi, and A. Selamat, "ELMS–DPU Ontology Visualization with Protégé VOWL and Web VOWL," *J. Adv. Res. Dyn. Control Syst.*, vol. 11, pp. 478–85, 2019.
[40] D. A. Zebari, H. Haron, S. R. Zeebaree, and D. Q. Zeebaree, "Multi-Level of DNA Encryption Technique Based on DNA Arithmetic and Biological Operations," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, 2018, pp. 312–317.
[41] Z. N. Rashid, S. R. Zeebaree, and A. Shengul, "Design and Analysis of Proposed Remote Controlling Distributed Parallel Computing System Over the Cloud," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 118–123.
[42] S. R. Zeebaree, A. B. Sallow, B. K. Hussan, and S. M. Ali, "Design and Simulation of High-Speed Parallel/Sequential Simplified DES Code Breaking Based on FPGA," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, 2019, pp. 76–81.
[43] M. Y. Kashmola and A. B. Sallow, "Information Hiding Techniques Using Network Protocols," *AL-Rafidain J. Comput. Sci. Math.*, vol. 8, no. 2, pp. 35–50, 2011, doi: 10.33899/csmj.2011.163650.
[44] N. O. Y. Subhi R. M. Zebari, "Effects of Parallel Processing Implementation on Balanced Load-Division Depending on Distributed Memory Systems," *J Univ. Anbar Pure Sci.*, vol. 5, no. 3, Art. no. 3, 2011.
[45] A. M. Abdulazeez and S. R. Zeebaree, "Design and Implementation of Electronic Learning System for Duhok Polytechnic University," *Acad. J. Nawroz Univ.*, vol. 7, no. 3, pp. 249–258, 2018.
[46] S. R. Zeebaree, H. M. Shukur, and B. K. Hussan, "Human resource management systems for enterprise organizations: A review," *Period. Eng. Nat. Sci.*, vol. 7, no. 2, Art. no. 2, 2019.
[47] A. M. Abdulazeez, S. R. Zeebaree, and M. A. Sadeeq, "Design and Implementation of Electronic Student Affairs System," *Acad. J. Nawroz Univ.*, vol. 7, no. 3, pp. 66–73, 2018.
[48] A. S. Y. Subhi Rafeeq Mohammed Zebari, "Improved Approach for Unbalanced Load-Division Operations Implementation on Hybrid Parallel Processing Systems," *J. Univ. Zakho*, vol. 1, no. (A) No.2, Art. no. (A) No.2, 2013.
[49] D. B. Abdullah and A. B. Sallow, "EOE-DRTSA: End-to-End Distributed Real-time System Scheduling Algorithm," *Int. J. Comput. Sci. Issues*, vol. 10, no. 2, p. 8, 2013.
[50] M. A. Mohammed *et al.*, "An anti-spam detection model for emails of multi-natural language," *J. Southwest Jiaotong Univ.*, vol. 54, no. 3, 2019.
[51] Z. N. Rashid, K. H. Sharif, and S. Zeebaree, "Client/Servers Clustering Effects on CPU Execution-Time, CPU Usage and CPU Idle Depending on Activities of Parallel-Processing-Technique Operations "," *Int. J. Sci. Technol. Res.*, vol. 7, no. 8, pp. 106–111, 2018.
[52] B. R. Ibrahim, S. R. Zeebaree, and B. K. Hussan, "Performance Measurement for Distributed Systems using 2TA and 3TA based on OPNET Principles," *Sci. J. Univ. Zakho*, vol. 7, no. 2, pp. 65–69, 2019.
[53] K. Jacksi, S. Zeebaree, and N. Dimililer, "Design and Implementation of LOD Explorer: A LOD Exploration and Visualization Model," *J. Appl. Sci. Technol. Trends*, vol. 1, no. 2, pp. 31–39, 2020.
[54] S. R. Zeebaree, "DES encryption and decryption algorithm implementation based on FPGA," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 18, no. 2, pp. 774–781, 2020.
[55] L. M. Haji, S. R. Zeebaree, O. M. Ahmed, A. B. Sallow, K. Jacksi, and R. R. Zeabri, "Dynamic Resource Allocation for Distributed Systems and Cloud Computing," *TEST Eng. Manag.*, 2020.
[56] A. B. Sallow, "Design And Implementation Distributed System Using Java-RMI Middleware," *Acad. J. Nawroz Univ.*, vol. 9, no. 1, p. 113, Feb. 2020, doi: 10.25007/ajnu.v9n1a550.
[57] H. K. Shaikha and A. B. Sallow, "Optic Disc Detection and Segmentation in Retinal Fundus Image," 2019, pp. 23–28.
[58] A. Sallow and D. Abdullah, "Constructing Sierpinski Gasket Using GPUs Arrays," *Int. J. Comput. Sci. Issues IJCSI*, vol. 11, no. 6, p. 131, 2014.

[59] D. Zebari, H. Haron, and S. R. Zeebaree, "Security Issues in DNA Based on Data Hiding: A Review," *Int. J. Appl. Eng. Res. ISSN 0973-4562*, vol. 12, no. 24, Art. no. 24, 2017.
[60] S. Zeebaree and I. M. Zebari, "Multilevel Client/Server Peer-to-Peer Video Broadcasting System," *Int. J. Sci. Eng. Res.*, vol. 5, no. 8, Art. no. 8, 2014.
[61] S. I. Saleem, S. R. Zeebaree, D. Q. Zeebaree, and A. M. Abdulazeez, "Building Smart Cities Applications based on IoT Technologies: A Review," *Technol. Rep. Kansai Univ.*, vol. 62, no. 3, pp. 1083–1092, Apr. 2020.
[62] I. M. Zebari, S. R. Zeebaree, and H. M. Yasin, "Real Time Video Streaming From Multi-Source Using Client-Server for Video Distribution," in *2019 4th Scientific International Conference Najaf (SICN)*, 2019, pp. 109–114.
[63] S. R. M. Z. Adel Al-Zebari Ali Selamat, "Electronic Learning Management System Based On Semantic Web Technology: A Review," *Int. J. Adv. Electron. Comput. Sci.*, vol. 4, no. 3, Art. no. 3, 2017.
[64] A. B. Sallow, "Android Multi-threading Program Execution on single and multi-core CPUs with Matrix multiplication," *Int. J. Eng.*, vol. 7, no. 4, p. 6, 2018, doi: 10.14419/ijet.v7i4.29340.
[65] C. A. Amira Bibo Sallow Mohammed Hazim Alkawaz, "Adoption of Mobile Augmented Reality as a Campus Tour Application," *Int. J. Eng. Technol.*, vol. 7, no. 4.11, pp. 64–69, 2018.
[66] S. R. M. Zeebaree, K. H. Sharif, and R. M. M. Amin, "Application Layer Distributed Denial of Service Attacks Defense Technique: A Reiview," *Acad. J. Nawroz Univ. AJNU*, vol. 7, no. 4, Art. no. 4, 2018.
[67] W. M. Abduallah and S. R. M. Zeebaree, "New Data hiding method based on DNA and Vigenere Autokey," *Acad. J. Nawroz Univ.*, vol. 6, no. 3, pp. 83–88, 2017.
[68] I. A. Khalifa, S. R. Zeebaree, M. Ataş, and F. M. Khalifa, "Image Steganalysis in Frequency Domain Using Co-Occurrence Matrix and Bpnn," *Sci. J. Univ. Zakho*, vol. 7, no. 1, Art. no. 1, 2019.
[69] S. R. Zeebaree and H. Rajab, "Design and Implement a Proposed Multi-Sources to Multi-Destinations Broadcasting Video-Signals," in *2019 4th Scientific International Conference Najaf (SICN)*, 2019, pp. 103–108.
[70] N. N. Qaqos, S. R. Zeebaree, and B. K. Hussan, "Opnet Based Performance Analysis and Comparison Among Different Physical Network Topologies," *Acad. J. Nawroz Univ.*, vol. 7, no. 3, pp. 48–54, 2018.
[71] O. Ahmed and A. Sallow, "Android Security: A Review," *Acad. J. Nawroz Univ.*, vol. 6, no. 3, pp. 135–140, 2017, doi: 10.25007/ajnu.v6n3a97.
[72] A. B. Sallow, M. Abdlqader, N. E. Tawfiq, and M. A. Shallal, "Initiating an Outcome-Based Education Environment at a Higher Education Institution: A Case Study," *Acad. J. Nawroz Univ.*, vol. 8, no. 3, pp. 39–49, 2019.
[73] A. Sallow, *A Fault Tolerance Distributed Real-Time System. Design and Implementation*. GRIN Verlag, 2014.
[74] H. Dino *et al.*, "Facial Expression Recognition based on Hybrid Feature Extraction Techniques with Different Classifiers," *TEST Eng. Manag.*, vol. 83, pp. 22319–22329, 2020.
[75] A. B. Sallow, Z. M. Taha, and A. S. Nori, "An Investigation for Steganography using Different Color System," *AL-Rafidain J. Comput. Sci. Math.*, vol. 7, no. 3, pp. 91–107, 2010.
[76] A. B. Sallow and Y. M. Younis, "Augmented Reality: A Review," *Acad. J. Nawroz Univ.*, vol. 8, no. 3, p. 76, Aug. 2019, doi: 10.25007/ajnu.v8n3a399.