IMPLEMENTATION OF CROSS PLATORM USING PUBLIC AND PRIVATE WEBOS

A thesis

Submitted to the council of

The college of science at university of

Sulaimani in partial fulfillment of the requirements

For the degree of master of

Science in Computer

By

Sara Adnan Anwar Sultan

B.Sc. Computer science (2007), University of Kirkuk

Supervised by

Dr. Kamaran HamaAli A. Faraj

Assistant Professor

December 2016 Reşemê 2416

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنكُمْ
وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ
وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ

صَدَقَ اللَّهُ العَظِيمُ

سورة المجادلة /الآية (١١)

Dedication

This thesis is dedicated to:

- Prophet of mercy to Mohammed “God’s peace be upon him"
- Whom I carry his name with pride "my father"
- The fountain of patience and tender "my mother"
- Who missed him in the face of difficulties and who taught me the road to success "my husband"
- The Hearts and good intentions of "my sincere sisters"
- Those are most beautiful and wonderful in my life "Anwar & Ali"

Acknowledgments

Thanks to Allah for giving me this opportunity, the strength and patience to complete my dissertation finally, after all the difficulties and challenges.

I express my sincere gratitude and appreciation to supervisor Dr. kamaran Hama Ali A. Faraj for invaluable advice and encouragement me during the realization of the research. Also, I sincerely thanks University of Kirkuk especially Dr.Omeed Al-Muhnds and Mr. Ahmed H. Hussein for providing me the opportunity to completes my MSc study. Also, I acknowledge Dr. Azad H. Fatah that carefully read dissertation and made the linguistic evaluation. I sincerely thank Mr. Saad Al-Jumaily at university of Tikrit and Mr. Bzhar J. Fatah for their support in many aspects; valuable advice.

Finally, my appreciation goes to my family for their patience and affection who helped me to complete my M.Sc. study.

ABSTRACT

The Web Operating System (WebOS) is a new generation of Traditional Operating System (TOS) that is used to facilitate and develop frameworks of a new paradigm by internet facility instead of being fixed to a particular machine (locality). It is an internet facility which user can access to own data remotely wherever by any computer where internet is available. WebOS is a new form of Cloud Computing -globally distributed computing which works on three service models of CC (Infrastructure as a service (IaaS), Platform as a service (PaaS) and Software as a service (SaaS). Furthermore, WebOS provides several functions of a TOS over the cloud of computers depending on the requirements of the Information Communication Technologies (ICT). In the proposed system WebOS designed by the web technology (XAMPP) that is a Cross platform, Apache, MySQL, PHP and Perl and investigated over two different types of ICT namely Private network-PrWebOS (intranet) and Public network-PuWebOS (internet). The number of users decided to use PrWebOS Wireless Local Area Network (WLAN) or PuWebOS by the two very common types of WebOS namely (EyeOS and Lucid) which are tested over PrWebOS and PuWebOS. In online, “it has an internet on the computer. So, WebOS is a computer on the internet”. But in a private, “it has a WLAN on computer on the intranet. So, WebOS is computer on the intranet. WebOS is exactly imitating interface of a TOS, even WebOS doesn't act together directly with the computer's hardware, but the client computer must still have a TOS. Nevertheless, the reason behind the popularity of the WebOS that it is economic because it is to reduce the hardware and software costs significantly. The WebOS duties are storing too large amount of user's data in a very basic way with simple interface. Users can (share files, listen to music, read news and etc.) depending on the diversity and richness of the WebOS used by the user.

Contents

Chapter one: General Introduction

Chapter Two: Theoretical Concepts

Chapter Three: The proposed WebOS

Chapter Four: The Implementation of proposed system

Chapter Five: **Conclusions and Suggestions**

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Description
Apache	extensive patching of existing code
Ajax	Asynchronies JavaScript and Xml
CC	Cloud Computing
ChWebOS	Chain web operating system
COS	Cloud Operating System
CPU	Central Process Unit
CRM	Consumer Relation Management
C/S	Client / Server
DB	Data Base
EyeOS	Eye Operating System
etc.	etcetera
FIFO	First In First Out
FTP	File Transfer Protocol
GUI	Graphical User Interface
HTML	Hyper Text Mark-up Language
HTTP	Hypertext Transfer Protocol
IaaS	Infrastructure as a Service
ICT	Information Communication Technology
IIS	Internet Information Services
IP	Internet Protocol
IT	Information Technology
LAN	Local Area Network
LAMP	Linux, Apache, MySQL, PHP

Mac OS	Macintosh Operating System
MAMP	Macintosh Apache MySQL PHP
MySQL	My- Structure Query Language
OS	Operating System
PaaS	Platform as a Service
PCs	Personal Computers
PHP	Hypertext Preprocessor
PrWebOS	Private Web Operating System
PuWebOS	Public Web Operating System
QoP	Quality of Performance
QoW	Quality of Website
QoS	Quality of Services
RCU	Resource Control Unit
RR	Round Robin
RRCU	Remote Resource Control Unit
SaaS	Software as a Service
SOS	Server Operating System
SQL	Structured Query Language
S2C	SaaS 2Consumer
TBisz	Traditional businesses
TLT	Traditional Load-Time
TOS	Traditional operating system
UI	User Interface
VLT	Virtual Load Time
VOIP	Voice over Internet Protocol

WAMP	Windows, Apache, MySQL, PHP
W-LAN	Wireless Local Area Network
WWW	World Wide Web
WebOS / WOS	Web Operating System
WEBtop	Web Desktop
XAMPP	cross platform, Apache, MySQL, PHP, Perl
XML	Extensible markup language

Chapter One
General Introduction

1.1 Preamble

Server operating system (SOS) and Web Operating System (WebOS) are structures to enable a new paradigm for/by the internet facilities as middleware over any types of networked communicate client server based. [VBEYACD97].

The networked communication type's examples are (Private WebOS) and (Public WebOS) [FASNN15]. The proposed system is mainly focused on the (PrWebOS and PuWebOS). The use of software without installation in any Personal Computer (PCs) is a miracle because cloud computing makes it possible in digitalized today's world. It saves both primary and secondary memories in your computer because data is on centralized data center located outside your own place [JDY11]. The attractiveness of the WebOS is instead of being fixed to a single location for example, the PCs services can dynamically organize responsibilities onto internet computing resources-client. The term of WebOS is a software platform that interacts with the user through a web browser and does not depend on any particular (TOS) [Sha13].

In everyday life, people must often rely on computer to sort out most of daily activities. It is impossible to any computer to work without software -TOS. The weakness of the TOS is due to its availability (i.e. it's limited locally and TOS doesn't have the high level of availability as it is available on only one computer). While the utilization of World Wide Web and the internet over networks were rapidly growing, the TOS became WebOS and its advantages of locality TOS desktop services dynamically work onto internet computing

Because of several factors, cloud computing concepts become popular which can be summarized as follows:

1. Increasing of high-speed and better availability of internet technologies; during the past few years.
2. In cloud computing users work with Web-based (WebOS) rather than local (TOS), storage and software.

WebOS is not the replacement of TOS but it's enhancement of TOS [SSK15].However TOS and WebOS complement each other. It’s impossible to any computer to work without TOS.

Operating System (OS) is a special kind of program that organizes and controls computer hardware and software. OS interact directly with computer hardware and serve as a platform for other applications. Whether it's Windows, Linux, UNIX or Mac OS, computer depends on its OS to function but a WebOS is a user interface (UI) that allows users to access applications stored completely or in part on the Web [SSK15]. The OS of WebOS that is installed in the server is a main, but the TOS that connect to WebOS via internet is a secondary OS.

Nowadays, the hot topic that rapidly grows and becomes very popular in the subject area of TOS is a WebOS. The implementation of WebOS is built on the internet facility and distributed computing. The objective of this WebOS is to deliver the full benefit of the world wide area (WWW). The WebOS have includes mechanisms for persistent storage, remote process execution, client management, authentication and security [Tri14].

The expectation of the proposed system tests will be very clear and show that there is a direct relation between the general dimension criteria for measurement evaluating and quality of service.

The dimensions of the criteria are content, design, organization, and user - friendly qualities.

These dimensions together with their comprehensive indicators and check list can be used by web designers and developers to create quality websites to improve the electronic service [HA11].

One of the WebOS type is Lucid desktop (formerly known as the Psych desktop) which is built on a base of PHP and is a presume-oriented web desktop service. This desktop can be installed on a web server like EyeOS, and is remarkably simple to use [Sha13]. Nevertheless, the EyeOS is an open source web desktop following the cloud computing concept it is mainly written in PHP, HTML. It acts as a platform for web applications written using the EyeOS toolkit. It includes a desktop environment with applications and system utilities. It is accessible by portable devices via its mobile front end. The examples of portable devices of front end are (the smart devices such as laptop, I Pad and mobile).The new contribution is to investigate better performance between methods namely (PrWebOS) and (PuWebOS).

Generally public and private WebOS are types of cloud computing. The services provided by a **public cloud** are offered over the internet and are owned and operated by a cloud provider. Some examples include services aimed at the general public, such as online photo storage services, e-mail services, or social networking sites and services for enterprises can also be offered in a public cloud. Nevertheless, in a **private cloud**, the cloud infrastructure is operated solely for a specific organization and is managed by the organization or a third party [The16].

The cloud computing has three forms of services models which are: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). In SaaS, the user uses the collections of application running in the cloud, while in PaaS the user uses the tools provided by the platform from classes' libraries and other language supports. However, the user has control over the operating system and the application deployed in IaaS with changing the Infrastructure components and configurations [Abu14]. In the proposed system the PrWebOS focused on IaaS type of cloud computing because the operating system is an infrastructure and it is similar to TOS. WebOS is mainly focused on enabling different kinds of phone calls, messaging services, settings and device information, multitasking and synchronizing phone book data and other virtual resources [ST11]. WebOS is one of cloud computing implementation [Tri14] and also it is one of the most recent and hot computer science topics by ICT. The WebOS is presented for CC and it is potential assessed when compared with TOS, The proposed system designed by Web technology (XAMPP) with the database backend-MYSQL, Middleware with PHP, and Frontend- Html.

1.2 Components of WebOS

Before Internet revolution, the TOS were in the traditional style and the TOS were in a very low stage compared to new versions of TOS namely WebOS. The unavailability, poor quality of service and locality of TOS is due to the very primitive communication facilities. At that time, the only way to install TOS was traditional and very limited (i.e. each computer 100% relay to own TOS), however WebOS provides all the facilities and more UI is exactly similar to the TOS as features of multitasking and multiuser methods. The users can use different applications in the same time and it is a multitasking method. The multitasking means running multiple applications by CPU simultaneously. Both of PrWebOS and PuWebOS are multiuser methods due to several users can connect to WebOS server in the same time to access of web applications.

The cloud user is an interface that is user-friendly more straightforward and with more efficiency [CPWYCH12]. Nevertheless, the increasing use of high-speed internet technologies, the concept of cloud computing has become more popularity especially economically.

The cloud services can provide speed, efficiencies and cost savings. In CC the users work with Web-based rather than local storage and software [KM13]. The costs save means that the users do not need to buy extra software and hardware.

There are three major components of WebOS which are:

1. **Graphical User Interface (GUI):** It's provides an interface which the user interact with the system and it is divided into three parts, the profile editor, the resource editor and the request menu. The resources are described through a profile .The profile editor helps the user to generate a new profile of resources he wants to make available for other users. The resource editor provides means by which resources can be selected and used. The third part of the GUI, the request menu, provides an easy - to- use interface to resources of WebOS Net [KA05].

2. **Rresource Control Unit (RCU):** it's accepts the service request from the user interface and contacts several known warehouse to find WOS node, where the requested service can be executed. The first thing that happens is that the local warehouse is contacted ,Then other known warehouse in the WOS Net .if an answer is found the RCU asks for the service execution and returns the result to the user. After successful execution, the local warehouse will be updated [KA05].

3. **Remote Resource Control Unit (RRCU):** It accepts a service request from other WOS nodes and examines whether the execution is allowed or not. Therefore, the resource warehouse is accessed. RRCU transmits the answer to the client-side RCU, the service execution itself is also managed by the RRCU, which contacts the resource warehouse for second time to verify access rights. The service is executed and the result is passed to the client- side RCU [KA05].

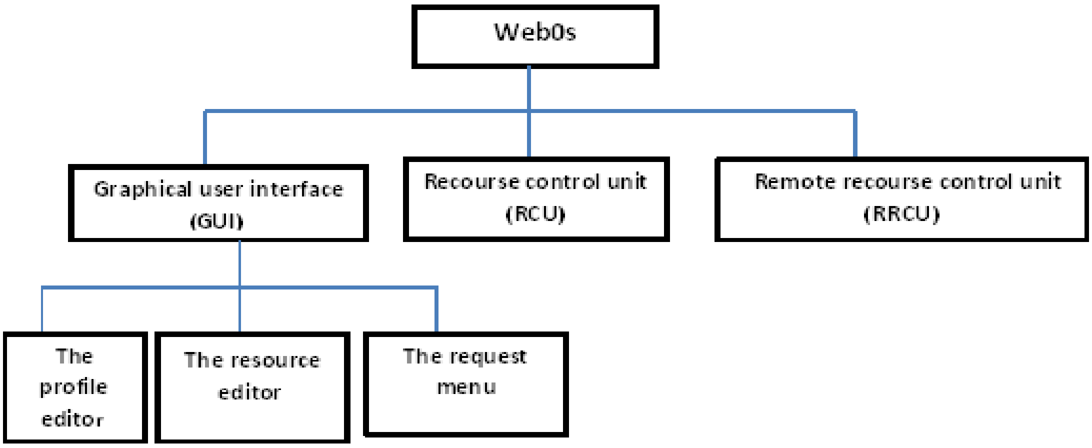

Web0s
Graphical user interface (GUI)
Recourse control unit (RCU)
Remote recourse control unit (RRCU)
The profile editor
The resource editor
The request menu

Fig (1.1) The main components of WebOS

1.3 Related work

1- Vahdat, Belani, Esatham, Yoshikawa, Anderson, Culler and Dahlin (1998) [VBEYACD98], demonstrate the power of providing a common set of the operating system services to wide- area applications, including mechanisms for naming, persistent storage, remote process execution, resource management, authentication and security. On a single machine, application developers can rely on the local operating system to provide these abstractions. In the wide area, however, application developers are forced to build these abstractions themselves or to do without. This ad-hoc approach often results in individual programmers implementing non-optimal solutions, wasting both programmer effort and system resources.

2- Peter G.Kropf (2000) [Pet00], discussed the WOS approach and its clarified components and starting by describing the general approach taken in the WOS project to define system services for global computing and he define the WOS as a framework consists in the total of the interactions between nodes relying on the basic communication protocol and multiple warehouse on each node, it designed as a completely open system to every user.

3- Mohit Jain, Mohd Danish and Hemant Yadav (2011) [JDY11], they examined the cloud computing , and referred to the following remarks:

   A. The concept of cloud computing is known as internet based computing with share resources, software and information is provided to computers and other devices.

   B. They clarified the implementation of cloud computing (Saas like salesforce.com , Paas like google app and Iaas like go grid) also cloud computing in which services and software are provided
   Over the internet or cloud which is very cheap and affordable.

4- Rajesh Sharma (2013) [Sha13], explored the WEBtop is a network application system for integrated web applications into web based work space. As a case study, the WebOS is evolving as a form at rapid pace, promising to free us from windows once and for all. They referred to some benefits of WebOS and the notion of the WebOS attempts to unify web applications to some extent. And in this paper he has reviewed some most interesting WebOS available nowadays and tried to provide a detailed description of their features.

5- Ananya Tripathi (2013) [Ana13], introduced that the WebOS is user interface that allows users to access to desktop and applications installed on the web, it may be similar to the user interface of Traditional Operating system like windows or Linux, but it does not directly interact with the computer’s hardware resources .WebOS reasons for online OS gaining rapidly popularity among its users are reliability, availability, fault tolerance, load sharing and function sharing by the users.

6- Hesham Abusaimeh, (2014) [Hes14],offered a brief overview about cloud WebOs and defined the cloud computing as the most efficient technology that reduces the time , cost and resources which are used by (IT) companies of any size from small to large companies.

7- Rajesh Sachdeva, Preeti Sharma and Neha Kataria (2015) [SSK15],in their paper described WebOS that is a newly emerging operating system that accessible via browser,with multiple integrated built in applications that allow the user to easily manage and organize / his/her data from any location. They stated that it has been called WebOS because it is present on the web and not on the computer of the user, all data that used are being stored on server of the WebOS provider. And they are referred to the two most popular approaches relying on flash technologies and XML (Ajax) technologies.

8- Therrien (2016) [The16], discussed the Cloud services which are typically made available via a private cloud, community cloud, public cloud or hybrid cloud. Generally speaking, it has also been services provided by a public cloud are offered over the Internet and are owned and operated by a cloud provider with some examples including services aimed at the general public, such as online photo storage services, e-mail services, or social networking sites. However, services for enterprises can also be offered in a public cloud. In a private cloud, the cloud infrastructure is operated solely for a specific organization, and is managed by the organization or a third party. In a community cloud, the service is shared by several organizations and made available only to those groups.

1.4 Problem Statement

This thesis focuses on the comparison between TOS and WebOS concerning the limitless of; performance, efficiency, distance, reliability, portability, cost compatibility (lack of hardware and software), capacity and computer distributed. The main aspire of this system is to modify the traditional locality of operating system to global electronic operating system. In order to access any application in WebOS is only need of a web browser and internet connection.

This thesis tries to find a solution modified from commercial to open source.

Thus, the WebOS brings cost down because it does not need license of software frequently and buying new hardware for a (PCs) owner. Also, the proposed WebOS is the solution of possibly increasable overlap functionality of TOS. The TOS features are in WebOS (i.e. multiuser, real time and multitasking) that are similar to TOS features.

The thesis will solve the real life and results transparency problem. The proposed system deeply concentrates to enhance of Platform as a Service (PaaS) over PrWebOS and PuWebOS. The PrWebOS is limited in distance having a high level of reliability and security comparing to PuWebOS. Nevertheless, the PuWebOS is load balance and more sharable with the risk of security. Finally, the proposed system realizes the suitability that return to different purpose.

1.5 Aims of Thesis

The main aim of this thesis is to introduce a new model for WebOS. The user interface of this software system allows both users and administrators to navigate the system easily to find all necessary applications relevant to their needs and careers. It's impossible without using of TOS of the client personal computers (PCs) becomes irregular. The WebOS is very compatible issues and is across platform (independent). The design implementation steps of the WebOS are given; the system should be unlimited for storage and distance. Both types of operating system are available in the proposed system. These two types are operating system traditional (TOS) and WebOS (electronic OS).

There are several aims of that research

1- To introduce a popular model of WebOS created an interface of user; this software system allows clients to share and use WebOS application in anywhere and at any time.

2- It is an operating system that runs in server on the web that created the environment in a user’s machine from an internet application, the WebOS applications run by user’s web browsers, since all browsers perform. The same (most of the time), all application must come from the internet, also the proven unlimited of distance and capacity are in easy way to communicate to specific server without carrying any (PCs).

3- Prepare a WebOS that is integrated with social media services and customize the GUI to fit the social media services.

4- Makes a manageable system that can give services and permission per users and groups also can be implemented by compilers and editors.

5- Offers all of the Microsoft offices applications and utilize for the system users and provide VOIP service for cloud user.

1.6 Thesis Layout

In addition to chapter one, the thesis consists of four other chapters:

Chapter (two) is entitled**” Theoretical Concepts”**. This chapter contains all of information necessary to understand some kinds of new technologies used in operating system. Also, some descriptions with explanations are given for all WebOS types, Cloud Computing models types by WebOS, WebOS architectures components-level, required tools for designing the proposed system, and the recommendation algorithms and techniques. **Chapter (three)** is entitled **“The proposed WebOS”**. This chapter describes the design and implementation steps of the proposed WebOS including the programming and the used tools. **Chapter (four)** is entitled **“The Implementation of Proposed System”**. The operations steps of proposed system are described in this chapter.

Chapter (five) is entitled **“Conclusions and Suggestions”**. In this chapter a list of derived conclusions and remarks is given, in addition to some suggestions for future works.

Chapter Two
Theoretical Concepts

2.1 Introduction

Over the past decade, technologies have made great progresses due to the extremely significant and revolutionary ICT facilities is undoubtedly the Internet.

The rapid development of networked and mobile computing, as demonstrated with the ever growing Internet has led to a global infrastructure, as well as to the introduction of new ICT functionality [Kro99].The Internet seems to be very similar, only extremely large distributed architecture –which shall be opened to a widely used by WebOS [UK99].The traditional method of TOS had been modernized by the emergence of ICT for example the Internet. In the past few years, the internet has considerably improved the face of TOS and the ways organizations about the TOS functionality. In the coming years, WebOS are expected to continue their explosive growth in the real life of computer science as a relation of Web technology and operating system.

These systems have good price-performance relations and can avoid high costs for the installation of parallel supercomputers. Modern load sharing and load balancing systems can enhance this relation once more [UK99].The WebOS functions are independent platform (cross platform) from any device with internet access.

Since cloud computing operates only with internet, this leads to the development of WebOS. The cloud computing makes collaboration easier and can reduce platform-incompatibility problems [KM13]. The WebOS run on the browser platform-independent, since browsers are built to work across different TOS. Figure (2-1) shows the WebOS platform independent whereas TOS is dependent. In a single machine client the TOS types for example (windows, Linux, Unix and Mac OS) must be selected only one by the user exactly before booting OS in a single machine, thus it's a dependent, but WebOS easily support every OS types without selection and it is independent.

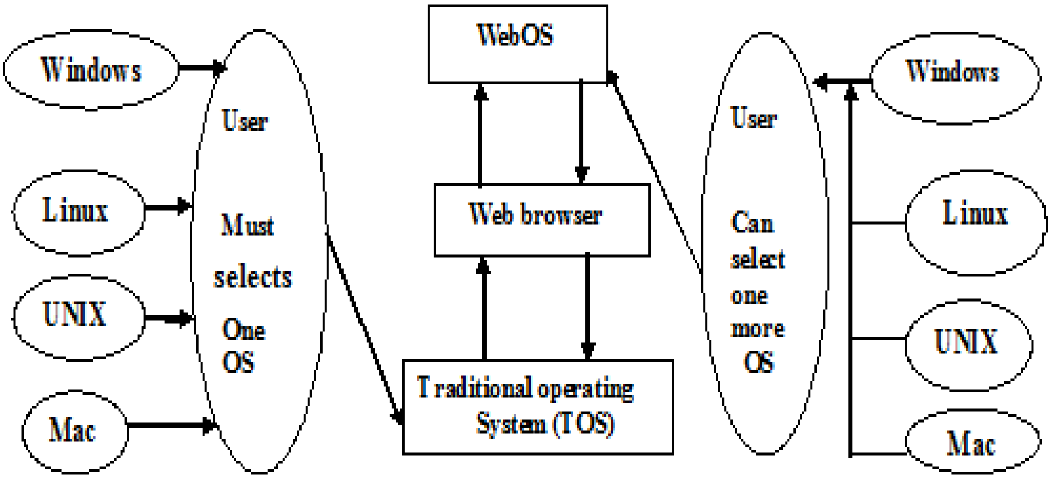


Fig (2.1) The WebOS platform independent and TOS dependent

Platform as a Service (PaaS) is one the three fundamental service model of cloud computing that our proposed system investigate to enhance the TOS functionality and etc. Also, is the delivery of middleware targeted towards developers with a platform that incorporates the entire development cycle, including hosting, testing, and deployment of web applications [DFFS12].

2.2 Traditional OS and WebOS

Before internet revolution, the (TOS) categories were in the traditional style; therefore the performances, quality of services and other factories were in a very low stage. The first known web-based operating system provided a platform that enabled user to benefit from computational potential of the web. WebOS provided users with plenty of tools through using a virtual desktop using the notion of distributed computing by the replicating its services among multiple interacting nodes to manipulate user requests.

One of the hottest topics that emerged these days between the area of internet and distrusted computing and the area of operating system is WebOS.

The WebOS will support geographically distributed, highly available, incrementally scalable and dynamically reconfiguring application. WebOS will include mechanisms for resource discovery, resource collaboration, persistent storage, remote process execution, resource management, authentication and security [MS02].As a result, from the preview ideas any attempt to design one single OS offering a fixed set of resource management function will have difficulty adapting to technological innovation or to new demands. Therefore, there is a need to the WebOS that make available to all sites on the network, the resources are available on that network to effect computation for which local resources are missing, these resources could be including processor speed, available memory or storage space and available application [KPU97]. From our experience;

1- TOS is a platform for software and hardware or it's a program that acts as in-between (i.e. users of computers and hardware of the computer). However, the WebOS is a computer of the internet or an internet service through which a user can access to a computer data remotely anywhere, anytime, by any computer, (any OS i.e. windows, Linux, etc.) as long as the internet facility is available.

2- The input, output and memory allocation in TOS are directly executed by the hardware, but in WebOS the input, output and memory allocation do not depend on special hardware for executing (i.e. impossible to achieve any activity in online WebOS without internet).

3- The directly interact of TOS with computer hardware and serve as a platform for any other application, but WebOS may be consider being similar to the user interface of TOS for example windows and Linux but it doesn't interact with any computer hardware resources.

4- The originally of TOS is to develop and provide a set of common system services and switches between hardware, central processor unit (CPU), hard drive and peripherals; but WebOS refers a delivery of computer resources over the internet instead of keeping data on the hard drive or updating application for user's need. Thus, no needs of Backup any more for WebOS.

5- Due to the globalized of WebOS the CPU scheduling types are impossible, but they are very common and important in TOS. The types of CPU scheduling in TOS are first in first out (FIFO), first short job (FSJ) and Round Rubin (RR) method.

6- Scheduler simulation waiting queue in TOS is very common and Figure (2.2) shows the scheduler simulation waiting queue in TOS. But there is no waiting queue in WebOS (i.e. private and public methods). There are a lot of comparisons between TOS and WebOS. TOS is still very useful because it is impossible to achieve any activity without the TOS; thus the WebOS is not replacement of TOS but both is complementing each other’s.

The old day Life without cloud computing: Traditional businesses (TBiz) by applications were very complex and costly.

The amount and variety of hardware and software required to run them are daunting. You need a whole team of experts to install, configure, test, run, secure, and update them. When you multiply this effort across dozens or hundreds of apps, it is easy to see why the biggest companies with the best IT departments are not getting the apps they need. Small and mid-sized businesses don’t stand a chance [TB11].

In order to design any WebOS, it is still necessary to rely on TOS. There should be more compare between WebOS and TOS but the most important for the proposed system evaluation is those compares that directly return to the OS (i.e. windows, Linux) and web (i.e. technologies, methods, middleware, web browser).

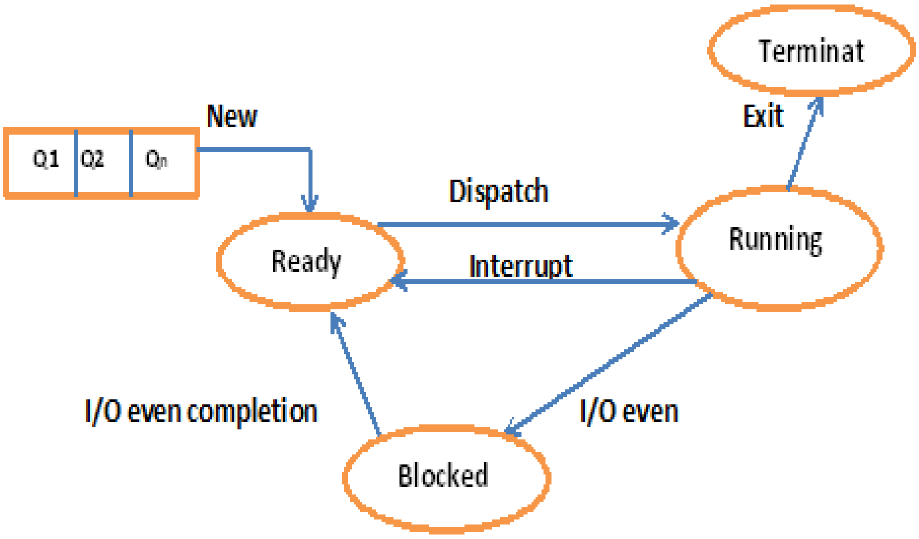


Fig (2.2) Scheduler simulation waiting queue in TOS

2.3 Technical Architecture of Cloud Computing

Web Operating System is a Web platform which allows the user to use a virtual Desktop through a Web browser on the Internet rather than using any particular local traditional operating system. This incredible technology allows user to access their own virtual desktop from anywhere around the world without even using a network with a remote PC.

In addition, the users are essentially using the Internet to work as a desktop, rather than an actually desktop computer. The kernel of a WebOS node is a system, a reactive system responding to requests from users or system. A WebOS-node integrates user, server, and applications [MS16]. The Figure (2.3) shows technical architectural of CC which is related to our proposed system.

The focal point of the proposed system is that it has three technical architectural layers which are platform (PaaS), infrastructure (IaaS) and service (SaaS).

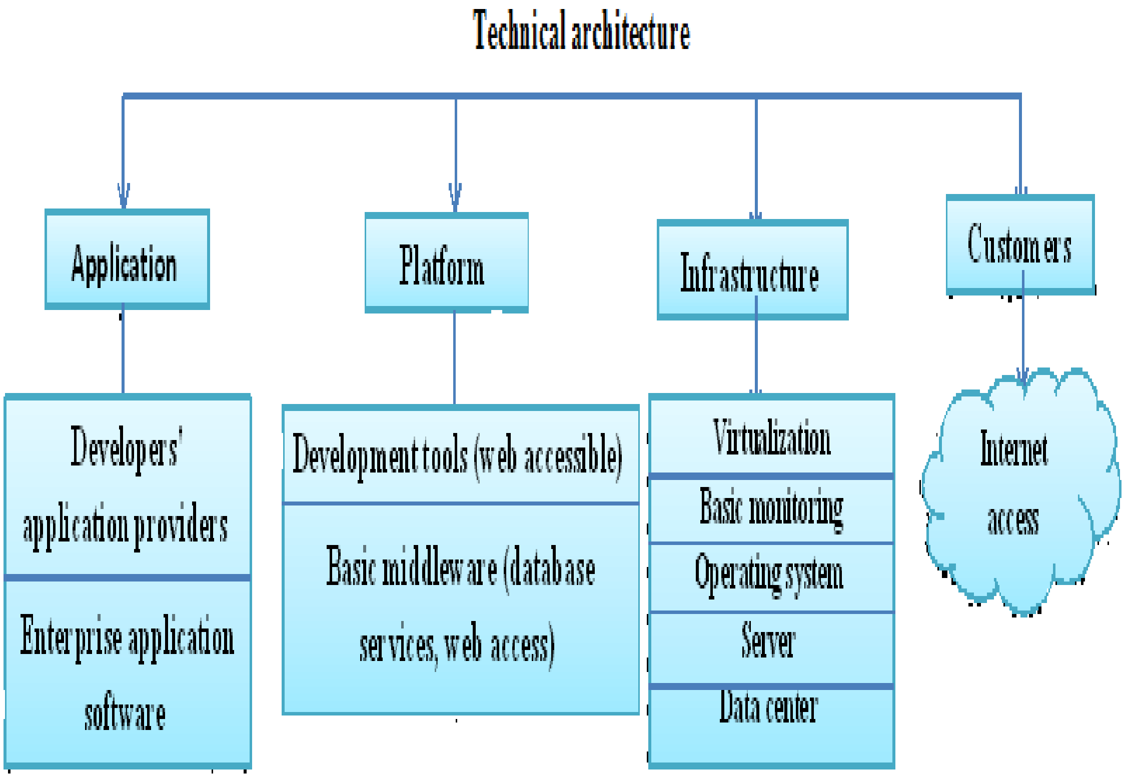

Technical architecture
Application
Platform
Infrastructure
Customers
Developers' application providers
Enterprise application software
Development tools (web accessible)
Basic middleware (database services, web access)
Virtualization
Basic monitoring
Operating system
Server
Data center
Internet access

Fig (2.3) Technical architectural of CC that related to proposed system

The main components of the CC services are to create a dynamic delivery services through the internet medium, as mentioned in the Figure (2.3) which demonstrate the main architectural structure of CC [KD10]. The components are given in the following sections: Application, Platform, infrastructure and customers. The direct CC value series in the respect of main architectural structure is shown in the Figure (2.4) for the CC service providing internet, IaaS, PaaS, and SaaS consultancy for main architectural structure purpose.

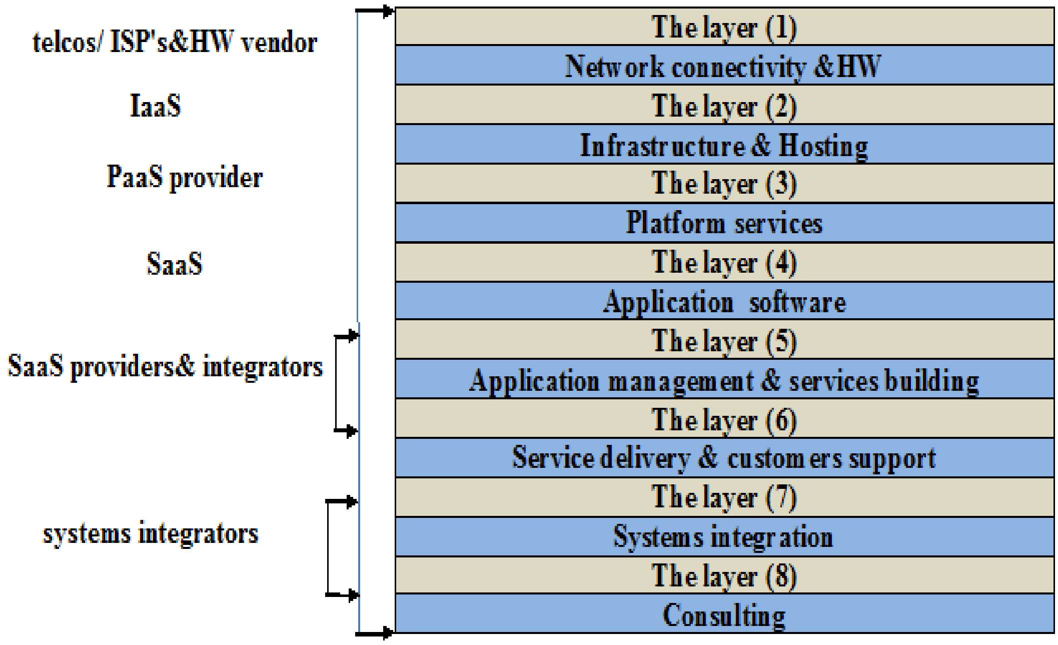


Fig (2.4) The CC services providing by internet, IaaS, PaaS, and SaaS up to consultancy for main architectural structure purposes

2.3.1 Application

Application is number four in CC architectural structure purpose as shown in the diagram of Figure (2.4).The Application in CC or WebOS is shared to all consumers and it is a special template by the WebOS that consumers can easily work on it. The application is a type of SaaS that consumers can easily open any application or games. Any change must be saved and stored in to WebOS-server if necessary. Typical consumer's 2 WebOS flow of information between consumers and WebOS usually is through the medium of Internet; see Figure (2.5).This flow may include application consumer- service request from consumers, application information, specifications or a list of services provided by businessmen etc.

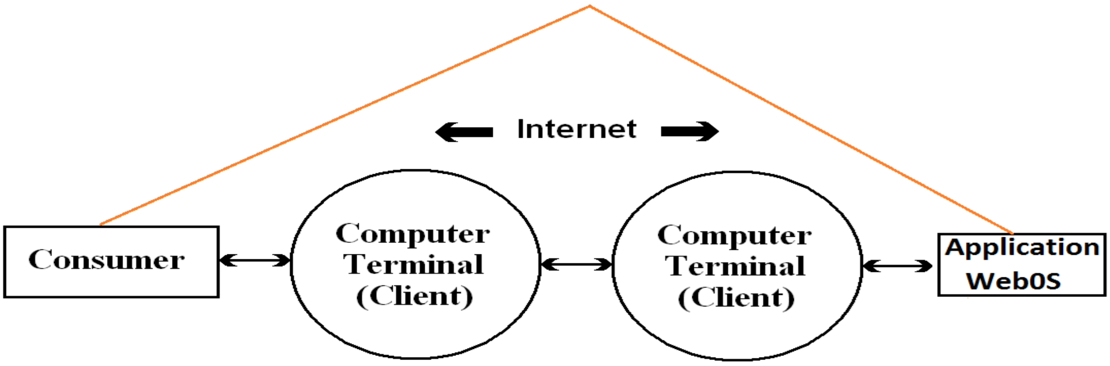


Fig (2.5) Consumers2WebOS flow of information

The application type is intangible because tangible products are all products that can be touched and delivered, but intangible products are all software that can be downloaded without need for delivering them. The example of the tangible product is a driving license that must be delivered by her/his owner. Both tangible and intangible products are utilized in our proposed system.

2.3.2 Platform

Platform is the layer number three in CC architectural structure purpose as shown in the diagram of Figure (2.4). Platform in CC and WebOS is shared to all consumers and it is a special template by the WebOS that consumers can easily work on it. The Window OS is not as good as the Linux OS. Since the Linux OS is the natural technology for enabling WebOS: its (modular, performance, power efficient, scales, open source and it's ubiquitous. Moreover, as the platform upon which the largest cloud infrastructures in the world have been built, Linux - unlike other available operating systems - has little left to prove as a component of cloud infrastructures, be they public or private. “Every time you use Google, you're using a machine running the Linux kernel [MCP09].

The WebOS is present on the web and not on the computer of the user, all the data are being stored on the servers of the WebOS provider [SSk15].with the capacity of storing large amount of consumer.

The platform provides a very simple and user friendly environment that users can share files, listen music, read news and can do many other stuff depending on the diversity and richness of the WebOS used by the user. Nevertheless, the cloud providers will often take the advantage of having the ability to modify the Linux source code in order to adjust and customize the kernel to their specific needs and hardware. For platform providers, Linux is the logical choice. Like the web architectures it is produce from, cloud computing platforms are often composed from many other open source projects, such as from databases to file systems to application and web servers to language runtimes. For cloud platform providers, however, the option to run non-commercial distributions does exist, and may be compelling. Platform providers will choose this path because creating a cloud infrastructure composed of thousands or tens of thousands of licensed nodes would be uneconomical with traditional per-server or socket models. By leveraging this lower cost approach, cloud providers are able to pass on the savings to customers. Besides its advantages in licensing, Linux is a more cost effective platform for providers to deploy and customers to target [MCP09].

In the proposed system the cloud provider is a company namely http://sarawebos.com that offers some components of cloud computing – typically Infrastructure as a Service (IaaS), Software as a Service (SaaS) or Platform as a Service (PaaS).The different types of OS is become the platform provider. The companies provide CC in order to achieve faster and error-free transactions, control and maintain limited inventory, load balance, etc.

2.3.3 Infrastructure

Infrastructure is the layer number two in the CC architectural structure purpose as shown by the diagram in Figure (2.4). Infrastructure and hosting in CC refers to the hardware and software components such as servers, storage, networking and virtualization software that is needed to support the computing requirements of a cloud computing model. In addition, cloud infrastructures include a software abstraction layer that virtualizes resources and logically presents them to users through programmatic means. One of the hottest topics that emerged these days between the areas of Internet as infrastructure distributed computing as Infrastructure and the area of operating system as platform is WebOS. Figure (2.6) illustrate the relation between TOS and WebOS. The hiring host is infrastructure as well [KD10].

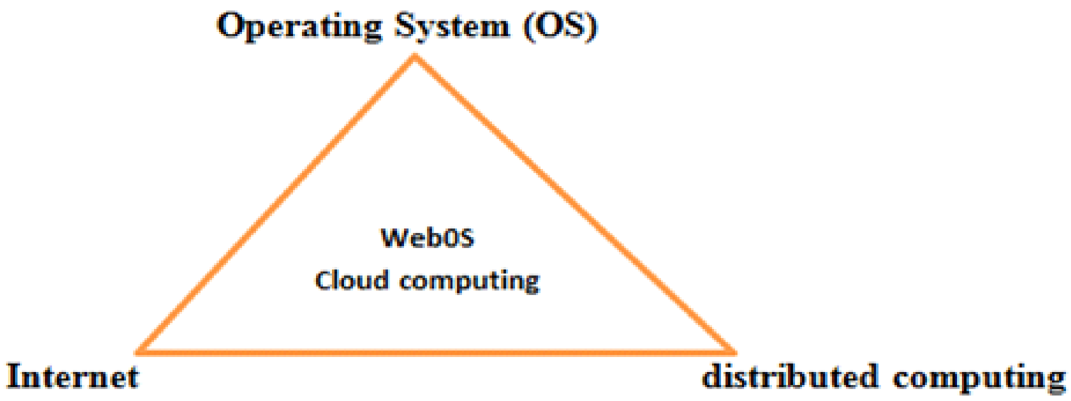


Fig (2.6) Elucidate the improvement of WebOS

While in PaaS, the user uses the tools provided by the platform from classes' libraries and other language supports. However, the user has control over the operating system and the application deployed in the IaaS with changing the Infrastructure components and configurations.

2.3.4 SaaS provider and Integrators

Application Management, Service Building, service delivery and cost support are the layer number five and six that are integrated in CC architectural structure as indicated in Figure (2.4). Application management service building, service delivery and cost support in CC or WebOS refers to the integrator of both numbers five and six. Both are application management and service building which are regarded as SaaS provider [KD10]

2.3.5 Systems integrators and Consulting

Systems of integration and consulting are the layer number seven and eight integrated in CC architectural structure purpose as shown by the diagram in Figure (2.4). The concentrating on elements (IaaS, PaaS and SaaS) as a standalone model, after mixture or combine between them is named integration and that could be a combine of mentioned elements for the CC service providing internet, IaaS, PaaS, and SaaS. Figure (2.7) shows standalone models of CC and combined models of elements. [KD10]. Accordingly, the companies that use standalone models achieve only one element of cloud computing for their business. For example Google usually provides SaaS and PaaS to its customers through Google Apps and Google App Engine individually. On the other hand, Salesforce.com provides PaaS and SaaS through its Force.com and Customer Relationship Management (CRM) in that order [KD10].

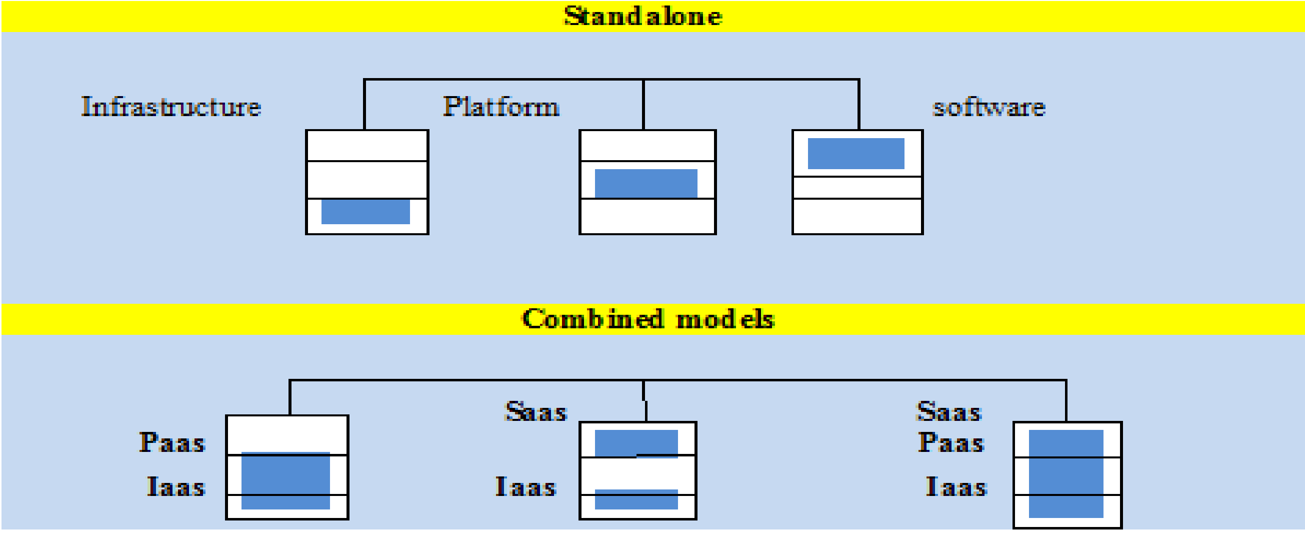


Fig (2.7) Standalone models of CC and combined models of elements

2.4 Models of WebOS

These models are composed of three service models including (variation related to, and four deployment models). The variation model related to is important for all types of cloud computing in real life reflecting the reality from the consumers' use of cloud clients to other parts.

2.4.1 Variation related to

Many models have been developed to handle WebOS requirements and needs, which are considered as parts of CC. The cloud clients are an important for entering to WebOS or all types of CC. The cloud clients could be web browser, mobile application, thin client, terminal emulator and etc. Figure (2.8) show the cloud clients components.

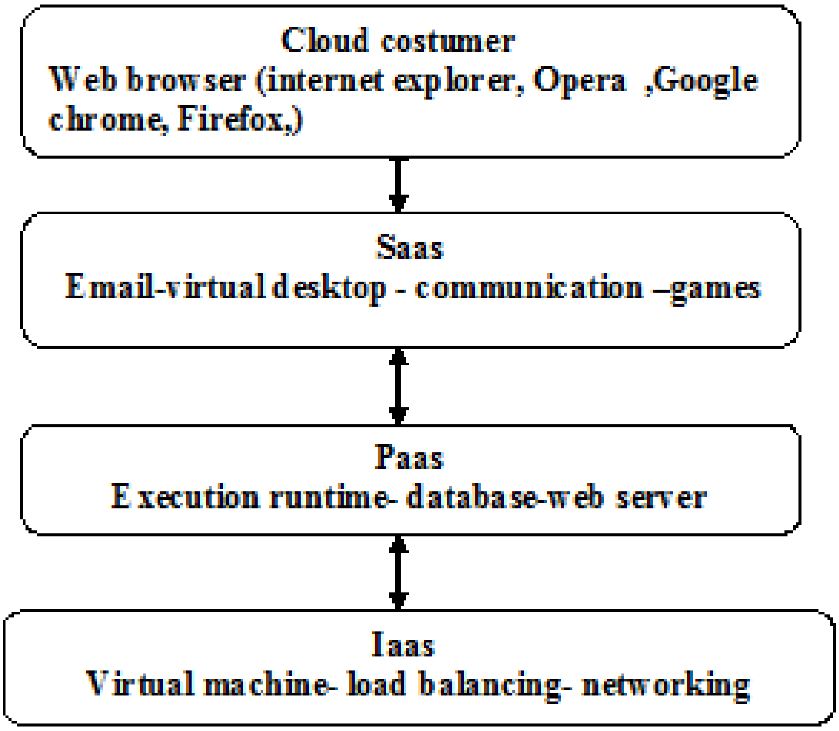

Cloud costumer
Web browser (internet explorer, Opera ,Google chrome, Firefox,)
Saas
Email-virtual desktop - communication –games
Paas
Execution runtime- database-web server
Iaas
Virtual machine- load balancing- networking

Fig (2.8) Cloud Client components

2.4.2 Deployment Models:

1) **Private cloud**: The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise

2) **Community cloud**: The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on- premise or off- premise.

3) **Public cloud**: The cloud infrastructure is available to the general public or a large industry group and is owned by an organization selling cloud services.

4) **Hybrid cloud**: The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by a standardized or proprietary technology that enables data and application portability(e.g. cloud bursting for load balancing between clouds). All deployment of cloud computing models [LOE11] are show in Figure (2.9)

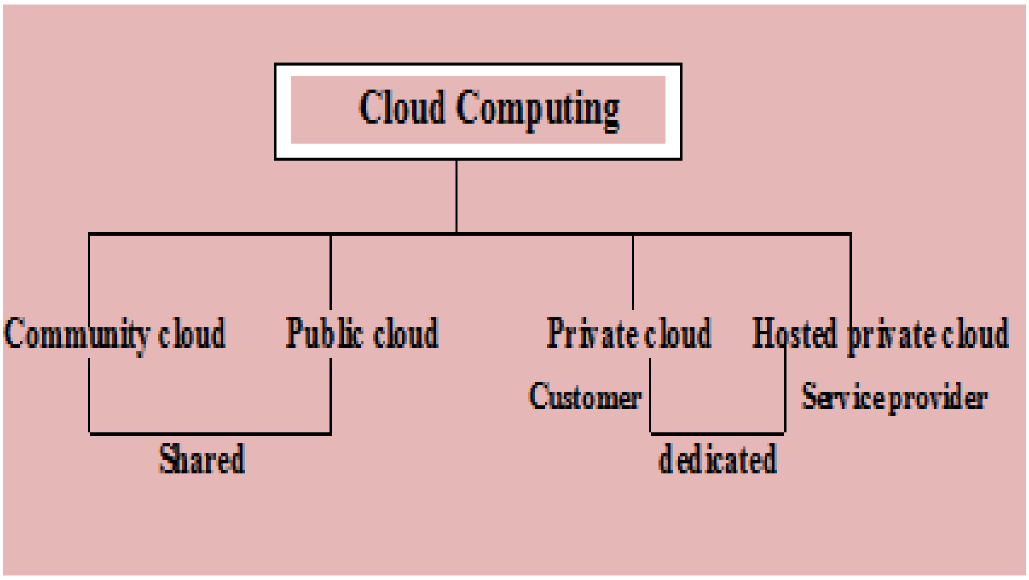


Fig (2.9) Cloud computing deployment models

2.5 WebOS

The model focal point of this models is to develop new evolving model of cloud computing which is called WebOS or Webtop [Abu14]. The Cloud is a simplified operating system that runs just a web browser, providing access to a variety of web-based applications that allow the user to perform many simple tasks without booting a full-scale operating system [san11]. Due to simplicity and working 24/7 of Cloud operating system WebOS the booting duration is very short and it can boot in just a few seconds from the Cloud. The user can quickly boot into the main OS, because Cloud continues booting the main OS in the background.

The main OS is a strong background for the WebOS. Combining a browser with a basic operating system allows the use of cloud computing in which applications and data "live and run" on the internet instead of the hard drive. Figure (2.10 A&B) shows the booting steps of WebOS and TOS.

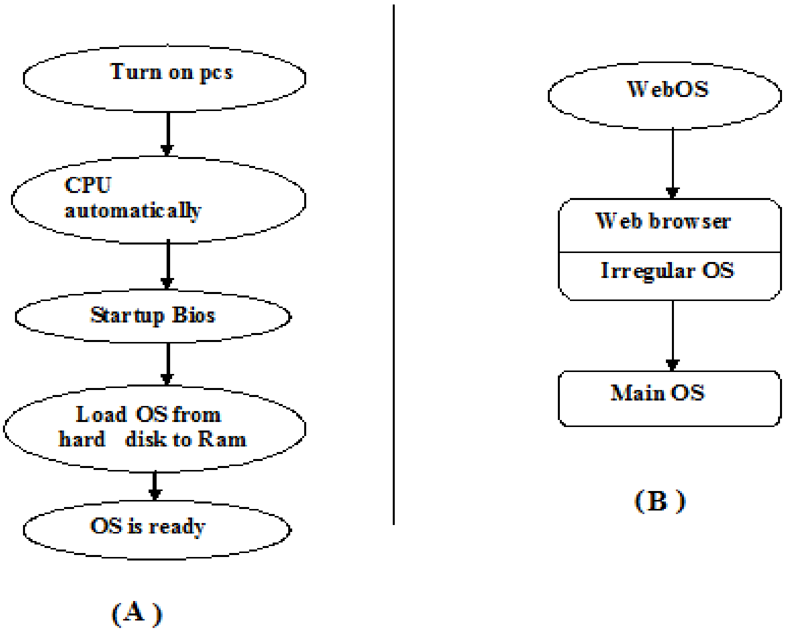


Fig (2.10 A&B) The booting steps of TOS and WebOS

The stable opinions of the significant real modification of (TOS) to WebOS are some similarities in the qualities and applications and assured result is how we work development system as cloud computing that eliminates some of the weaknesses that users suffer such problems when using operating system, We have reached the most of the good results and improved somewhat and was done in parts and applied in practice in both phases we came to the result obtained is solving the problem of distance and limited implementation of the requisite benefits some of the cost and increase the rapidity and improve efficiency of the TOS jobs.

(PrWebOS) is limited by the distance limitation signal of LP- link router. But, (PuWebOS) is unlimited of distance limitation signal. WebOS is a software platform that interacts with the user through a web browser and does not depend on any particular local operating system [Sud08].The client must have only web browser to work with WebOS and have access to own personal desktop, document, music, movies [PBDJSS10] and all etc. while Compared WebOS to TOS, the cloud technology offers numerous revolutionary advantages that can drastically improve the value creating capabilities of organizations. The most significant advantages are: much lower cost, faster time to market, and great opportunities for creating new sources of value [DFFS12].

2.6 The scope of WebOS

The WebOS has vast scope to do by clients:

1- Allow client to use web applications and run his request from anywhere anytime.

2- Storing the user's data without having to worry about the loss data.

3- Create new client interface (VD) of the system that similarity of TOS.

4- Popularity that have compatibility issues easy to install and implementation

5- Platform independence

6- Faster and safer access

7- Centralized storage

8- Huge data analysis

9- Data sharing

10- Better administration.

11- PC-centric

12- Robust and scalable

However, the scope of WebOS studies shall can't the following open- problems:

1- document-centric

2- Avoiding vulnerabilities of the systems

3- solve complex infrastructure

4- replacement and ignore TOS

2.7 Potential Benefits of WebOS

The advantages of WebOS based system could be summarized as below:-

A. ***Economy:*** the WebOS is much more economical than (i.e. TOS) regarding physical storage and services. Also users computer not necessary with a high-powered and high-priced to run cloud computing's web-based applications. While applications run in the cloud, not on the desktop PC, the (PC) does not need the processing power or hard disk space demanded by traditional desktop software. When users are using web-based applications the PC can be less expensive, with a smaller hard disk, less memory, more efficient processor and etc. In fact, the connected PC to the cloudOS does not even need a compact drive (CD) or digital video disk (DVD) drive, hence no software programs have to be loaded and no document files need to be saved. Obviously, the WebOS is very cost-effective comparing to TOS. Thus, as an alternative of pay for expensive software applications easily can get most of them in cloud computing for free such as the Google Docs suite and it is better than paying for similar commercial software.

B. ***Speed:*** time-booting and running are much faster in cloud because they have fewer programs and processes loaded into memory.

C. ***User friendly:*** the computers connected to WebOS or CloudOS can easily open, execute, configure, edit, delete, and update files online and the user of all the operating systems features and virtual desktop icons on the cloud.

It does allow the users to write documents, draw graphs, and compile classes and programs. [Abu14].

D. ***Improving consistency:*** It is portable, that attempts to unify our web applications to some extent, providing user with a single point of access, store files and return to from any computer on the planet. No hardware problems. Don't have need to back up or archive because the WebOS deals across platform it eliminates compatibility issues between the application and OS [PBDJTSS10].

E. ***Non maintenance required:*** There is no need to worry about maintaining and safety of your storage or servers, as cloud computing providers will take care of everything – out of sight and out of mind.

F. ***Effective file management:*** Rather than having your data scattered about the web with a vast range of services, the WebOS promises to make it easy to store, sort and access all of your files wherever you go from a single networked tool [Sud08].

G. ***Improving storages:*** WebOS users are not necessarily to back up or archive work because the WebOS provider's handles these functions coming up with a disaster recovery plan usually requires lots of cash and expertise businesses with small budgets can implement backup and recovery solutions that don't require a large up-front investment or third-party expertise.

H. ***Improving cross platform applications*:** open the web standard enables applications to run on multiple platforms because WebOS deals a cross platform, it eliminates incompatible issues between the application and OS.

I. ***Easier group collaboration:*** the sharing principal documents have a direct relation to better collaboration.

Most of the users ensure of this as an important advantage of cloudOS for example when the system crashes, without losing data or time: Just login in to Cloud OS from another computer and retrieve your work and continue working.

J. ***Increasing global reach:*** With WebOS, users from all over the world can apply immediately for using any application software to any organization at any time of the day.

2.8 Popularity of New Technology (WebOS)

All types of new technologies of CC services (i.e. WebOS) are popular because of:

A. *Time efficiency*

B. *Space efficiency*

C. *Superior working*

D. *Distributed efficiency*.

Time, space, hardworking, and distribution are four important factors terribly inefficient in TOS when compared to WebOS. WebOS is time saving system for both users and companies to exchange information between them. The advantage of WebOS is the cut time-to-load which is reduced all wasted time in the processes (i.e. scheduler simulation model is absolutely absent in TOS).

TOS is installed individually for each computer as a main and not irregular. Thus the main TOS in individual PC needs or occupies a large space, but space to establish WebOS needs only several computers connected to internet. So, WebOS needs much less space to establish than the traditional way because of the TOS clients are share recourse and becomes irregular OS. Also, the process of WebOS is much powerful, accurate and errorless if it is compared with the TOS. With the increasing use of high-speed Internet technologies during the past few years, the concept of cloud computing has become more popular. In cloud computing (CC), users work with Web-based, rather than local, storage and software. These applications are accessible via a browser and look and act like desktop programs. With this approach, users can work with their applications from multiple computers. In addition, organizations can more easily control data and reduce malware infections. Also, cloud computing makes collaboration easier and can reduce platform-incompatibility problems. Now, a growing number of organizations are adding to the cloud concept by releasing commercial and open source Web-based operating systems. It also includes many of traditional OS capabilities, including a file system, file management, and productivity and communications applications. In the case with Web-based applications, the WebOS functions across platforms from any device with Internet access [KM13].

2.9 Future of WebOS

WebOS is a technique of the future and it is a good chance to creating applications and technology more accessible than the past days.

The previous WebOS was bright, but the future of WebOS will be even much brighter. Nowadays, WebOS will expect to be a permanent solution for real life in communication organizations. Furthermore, the WebOS data centers will be automated and will maintenance accessible software architecture, WebOS users do not depend on software running on general local PCs but users depend on the software in WebOS. There are several factors to reach good future of WebOS.

1. ***Pre-emptive Application Monitoring:*** is a technology presently offered to use, nevertheless pre-emptive technology and software in soon future make this more robust and accurate. This will avoid interruption and make the company safer.

2. ***Technology to Confirm Uptime:*** a period of time when WebOS software is functioning and available for use by WebOS users. The WebOS users must have uptime guarantees of 99.9%.
3. ***WebOS Role in Disaster Recovery and Remote Access:*** the role of CloudOS in term of recovery of disaster. It enhances remote access and faster disaster recovery.
4. ***WebOS Becoming More Robust:*** WebOS is becoming more robust "fit and function" than about the "hype" surrounding the new technology. Most companies can benefit from WebOS, but some companies have suffered from their failed projects during the migration phase because the technology is still in the developmental stages. Typically, migrations fail because of inaccurate or missing data.
5. ***WebOS Ability to Validate Identities through Trusts:*** In the future, WebOS security systems will be able to validate identities through a "centralized trust." Identity-based security is thought to be more secure than current forms of security. More people will begin to trust WebOS when this happens.
6. ***Centralized Data is the Future in WebOS:*** Centralized data is expected in abundance in the future of WebOS. This allows companies to create huge databases. Patient care can be improved with centralized data in huge databases. Better stock market decisions are also possible.
7. ***More Capability:*** Mobile devices that rely on the cloud will become more powerful and thinner because all applications will be web-based. All mobile devices will store data that resides in the cloud and designers can add more capability and lower costs of the phone. One example of this concept is Apple's iCloud.
8. ***Hybrid WebOS Increases Efficiency:*** Hybrid WebOS is expected to help users and become more efficient by optimizing business process

performance. Businesses are excited about hybrid WebOS because it's enhances internal infrastructure and applications. The ability to scale the strength of local networks and WebOS is coveted by designers. Figure (2.11) shows Hybrid designers between PrWebOS and PuWebOS

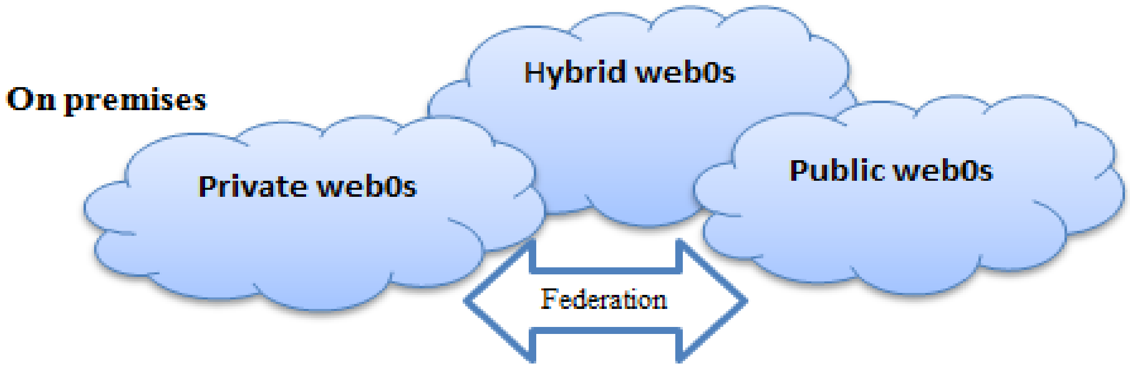


Fig (2.11) Hybrid WebOS between PrWebOS and PuWebOS

9. ***Commodity Hardware:*** low-cost hardware will make it easier to configure advanced data centers capable of complex algorithms at fast speeds. This will increase uptimes and improve user satisfaction.

2.10 WebOS terminologies

Figure (2.12) shows the main parts of adopted architecture of the established WebOS in this thesis. In the following, some WebOS basic terms are defined:

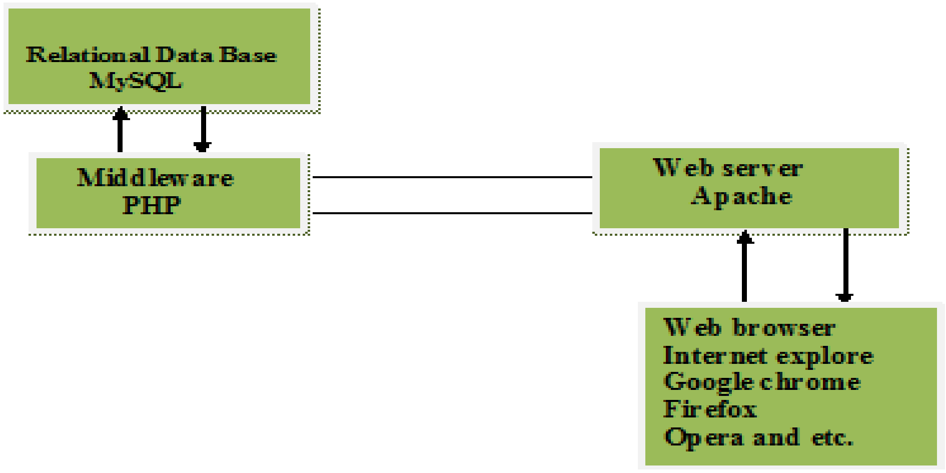


Fig (2.12) The architecture of the developed web based application

A. The Web

Normally WebOS users store their material on the WebOS server and retrieve information from it any time they want. WebOS consists of an enormous collection of documents in different formats, which are located on computers through the worldwide. Web communication medium consists of the following elements or collection of information and files that organized to webpages namely World Wide Web (WWW) and it's a subset of the Internet. The Web consists of pages that can be accessed using a Web browser. WWW is simply a Web in a way of accessing information over the medium of the Internet for information-sharing model and built on top of the Internet. The Web uses the HTTP protocol and it is a single language spoken over the Internet to transmit data [FM14]. Furthermore, Web services are using HTTP to allow applications to communicate in order to exchange business logic-Database web. The Web also utilizes browsers such as internet explorer, Firefox, opera and Google chrome in order to access web documents called Web pages. The web can be used by the users to upload and download files anytime by file transfer protocol (FTP). The web communication medium consists of the following elements, as explain in Figure (2.13).

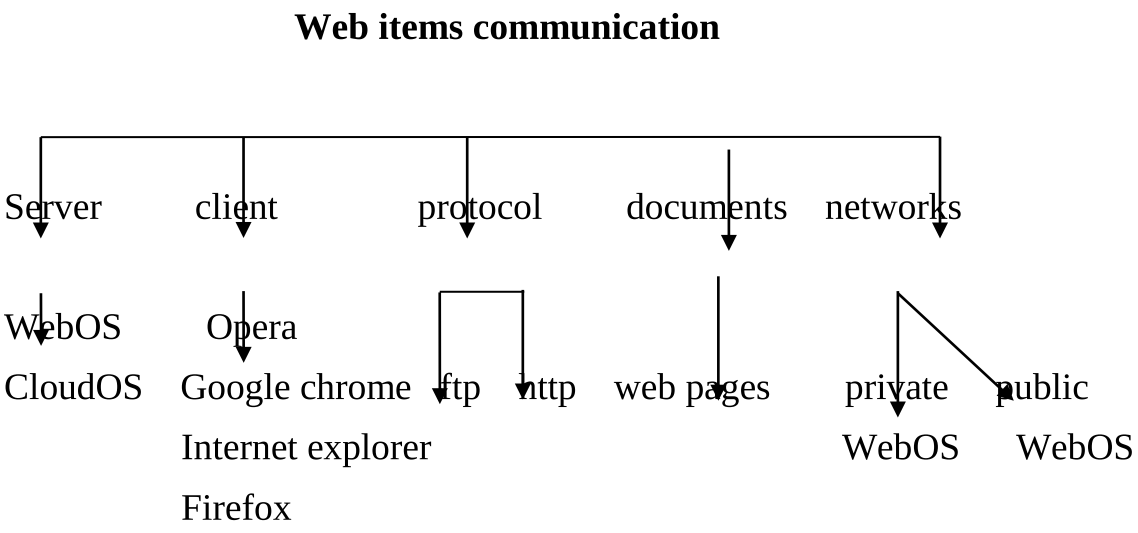


Fig (2.13) The web items communication

B. Web Server

A webserver is a program or software that can be used to deliver services to clients on request by using (HTTP). Services here include web pages in (HTML) language documents which may include additional contents like images, videos, style sheets and scripts [Nas13] .Furthermore, a web server is a specific type of server that knows how to communicate with clients using hypertext transfer protocol (HTTP). It is waiting for clients to connect and when a connection is established, they receive a request from the client and then respond—usually returning a document or image. The web server process is usually referred to as the Hypertext transfer protocol daemon or HTTPD [STE99].

The Client/Server (C/S) model is a system that performs both the functions of client and server so as to promote the sharing of information between them. It allows many users to have access to the same database at the same time, and the database will store much information. It has become so popular because it is being used virtually every day for different applications [Har14].There are two main types of Web servers such as Apache and IIS. The proposed system deals with Apache:-

Apache is a web server package that works under Linux as well as under other operating systems. The name of Apache comes from the concept of (extensive patching of existing code) and the primary advantage of Apache is generally free or available at modest costs [LFC99] and supports all majors of the OS.

C. Web Browser (Internet browser):-

It is a software application for retrieving, presenting, and traversing information resources on the World Wide Web. An information resource is identified by a Uniform Resource Identifier (URI) and may be a web page, image, video, or other pieces of content.

Although browsers are primarily intended to access the WWW, they can also be used to access information provided by Web servers in private networks [Bra11].

There are many web browsers that use from the users like

- Google chrome
- Internet explore
- Firefox
- Opera and etc.

D. PHP and MySQL

Hypertext Preprocessor (PHP) is an open source protector which means having access to source code and it can used altered and redistributed charge. The PHP originally is stood for personal home page. The PHP has high performance with GUI and suits to many different database system .The best to use database is MySQL which is very vast, robust, relational data base management system. Also it is enables the users to efficiently store, search, sort, and retrieve data. MySQL server is controls access client to data to ensure multiple users who can work with it concurrently.

One of the best features of both PHP and MySQL is that they work with any major operating system for example Windows, Linux, and Macintosh. The best webserver is apache will be:

1) window apache MySQL PHP (WAMP)
2) Linux apache MySQL PHP (LAMP)
3) Macintosh apache MySQL PHP (MAMP)

The following advantages of PHP and MySQL are strong motives behind their utilization in developing various web-applications:

- ***It is fast and easy:*** PHP has managed the perfect mix of power, and it is easy to use and learn, also PHP offers the best opportunity to develop powerful web applications, furthermore generalizations may become excellent reasons for choosing PHP.

- ***It is cross-platform:*** In the rundown of web architecture, PHP run on windows and UNIX and with both Internet Information service (IIS) and Apache. But the cross-platform abilities of PHP work as well on as wide variety of systems as any other available products for example Linux, UNIX and etc.
- ***It accesses everything:*** whatever the web-designer needs to use, it is more than likely that PHP has a built-in set of functions that make getting whatever the designer need very easy; that means it is more easier than other products and supports most of the platforms.
- ***It is constantly being improved:*** users employing the opened source development might be surprised by the quality of the software and variety of developers looking to improve the product. Also, MySQL is improving at amazing rates and works on many different platforms.
- ***It is free* to download** (without any cost): MySQL and PHP are completely free, that means users install and use it and pay nothing in the process. Free download or install makes the PHP become popular software. MySQL is a program that can store large amounts of information in an organized format that is easily accessible from scripting languages.MySQL software delivers a robust SQL (Structured Query Language) database server.
- ***It is cost effective:*** the oracle installation cost is incredible, there is no doubt that Oracle, Sybase, and Informix create terrific databases, but the cost involved will be prohibitive for many and MySQL free.

- ***It is quick and powerful:*** MySQL may not have every bell and whistle for relational database, but for most users, there is plenty (for example, creating a moderately size commerce site)

2.11 Structure model of WebOS

The structure levels of the cloud and the actors of the system which are: the Root and the normal user. The Root which refers to the admin as in UNIX, Linux world and the normal user who has certain permissions specified by the Root and can access his own desktop and launch his favorite applications. Both client users and root users can access the cloud WebOS after they logged in from any web browser. When the root is logged-in, he can get reports about users and monitor their actions via browsing log files and assigning permissions to the client users [Abu14]. Figure (2.14) show the structure models of WebOS.

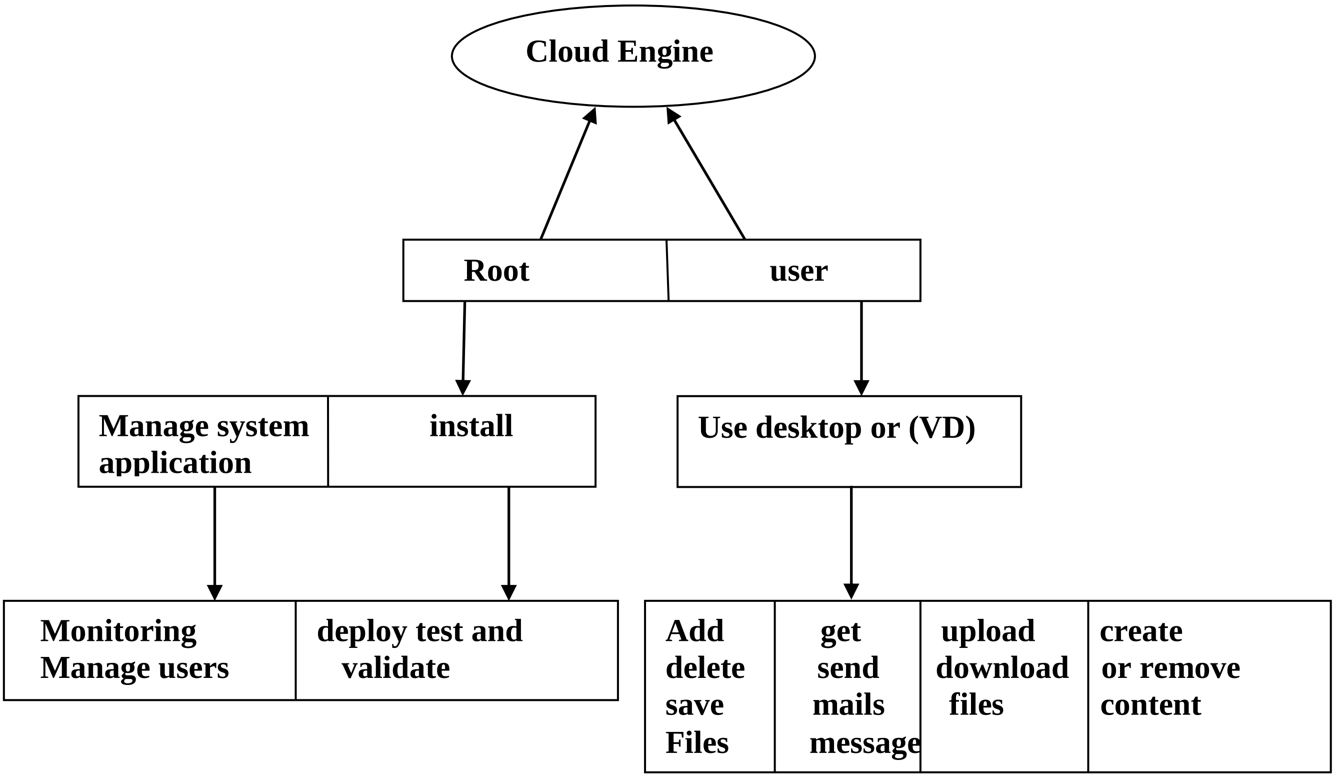


Fig (2.14) The structure models of WebOS

Chapter Three
The Proposed WebOS

3.1 Introduction

This chapter describes the design steps of the proposed WebOS. Generally, there are several types of WebOS, but the proposed system investigates only two popular types of WebOS (EyeOS and Lucid) Thus, both the mentioned WebOS types are tested on two methods which are PrWebOS and PuWebOS. PuWebOS is by the public hired domain with the name http://sarawebos.com for the purpose of investigation. Both of EyeOS and lucid systems are designed by cross Apache, MySQL, PHP and Perl (XAMPP).The designed EyeOS and Lucid run separately; one time over the Linux OS-PuWebOS and another time over windows- 7 OS- PrWebOS. However, the wireless PrWebOS distance limitation is up to hundred meters by the wireless LP -link router with the domain name of http://localhost/ as mentioned in previous section, the system was proposed to facilitate the implementation of the requests of users at the lowest cost, fastest time and without users need to carry PCs all the time as this system facilitates the pass process at anytime and anywhere. WebOS users can use the web services, applications - SaaS and PaaS. Also just need any type of web browser in client side with availability of internet-IaaS access. WebOS is a solution of some TOS difficulties or problems. The WebOS can be achieve all TOS functionality but it still not replacement of TOS because they are dependent on each other in order to finish process of WebOS, that means the Background of WebOS is TOS. A compromise between the PrWebOS and the PuWebOS is a combine of both WebOS methods in order to enhancee with less complication. The two methods are combined and create **"Chain web operating system"**

(ChWebOS) which is an appropriate and new contribution that found out in our investigation. Figure (3.1) show the cloud computing models and figure (3.2) presents the proposed system of WebOS.

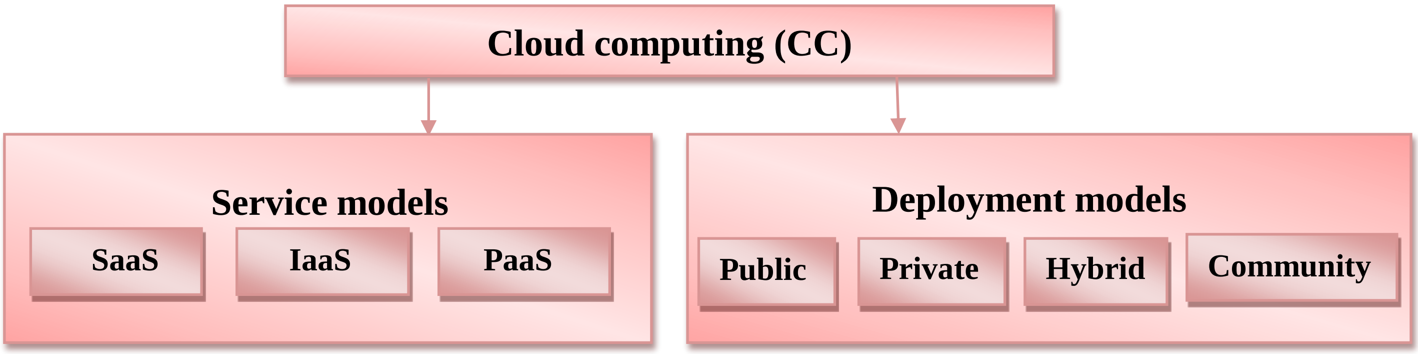


Fig (3.1) the cloud computing models

However, WebOS is new concept of Cloud computing so, its deals with cloud computing models as mentioned. In the proposed system have three types of deployment models that shown in figure (3.2).

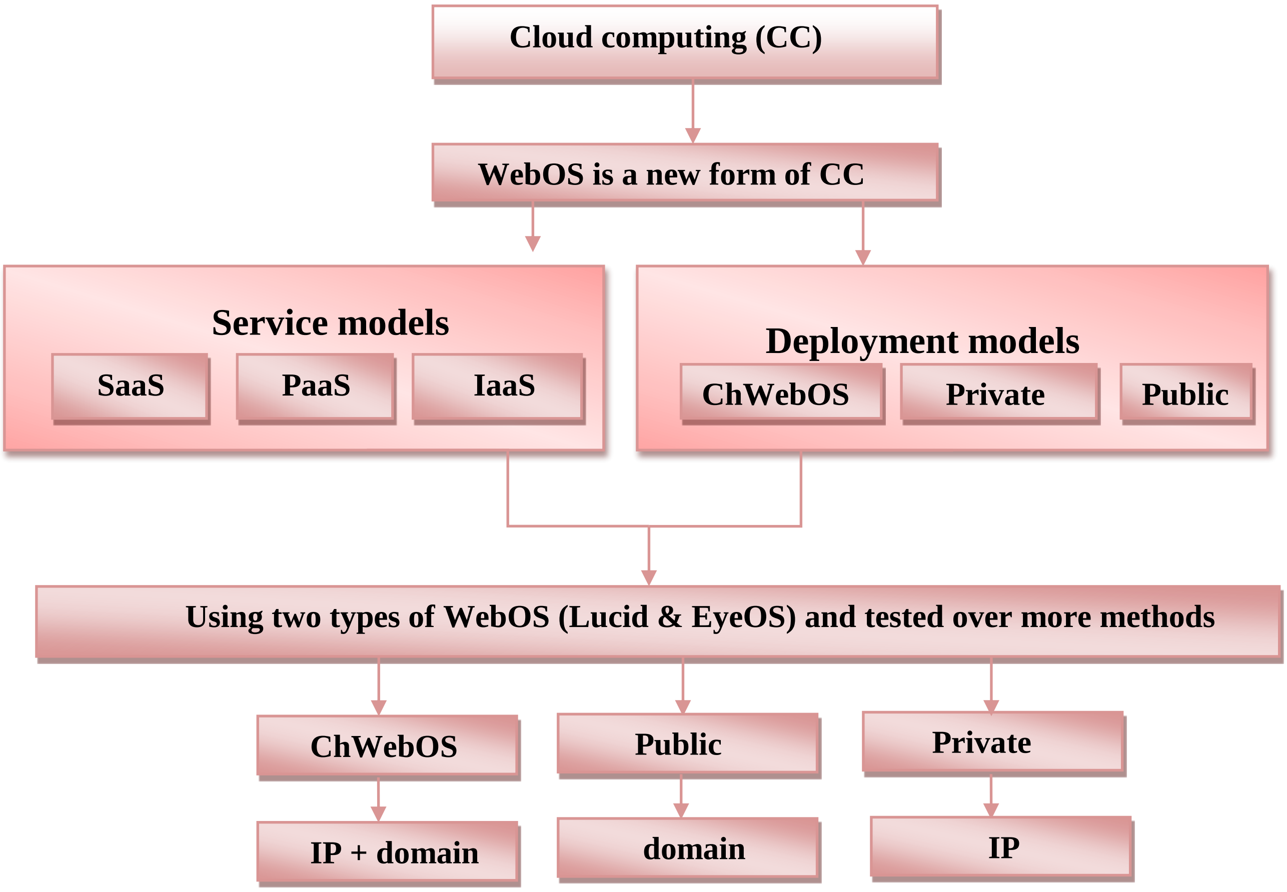


Fig (3.2) The proposed system layout

3.2 Architecture of the Proposed System

In general, the proposed system is designed to run either on PrWebOS or PuWebOS. Both methods are client/server based computers network. Nonetheless, the TOS in a single computer is not a C/S. PrWebOS and PuWebOS architectures are on C/S base networked computers with any network scales. The Blended PrWebOS and PuWebOS to test proposed system create (ChWebOS).

There are several areas of concern of PrWebOS and PuWebOS systems:

1. Back-end (IaaS).
2. Web application Data (SaaS-PaaS).
3. Front-end (SaaS).

The back-end is a backbone of any new technology system such as (WebOS). The developers should be sure that the established back -end is secure and is working faultlessly. The step that has highest priority in the system establishment schedule is the design of back-end module, because most of the errors in the systems return to the inaccuracy of back-end module. MySQL is an outstanding tool to implement the database system. The technology used in the proposed system to link the database to front- end interface is PHP with Window -7 OS as a server using of XAMPP application for the PrWebOS and LAMP - Linux. Figure (3.3) shows the PrWebOS and PuWebOS in different OS. The Notepad$^{+ +}$ has been used as an editor for typing codes of the proposed system. Almost all of the work of WebOS takes place on the server, except the specific application called "webserver", which is responsible for communication with the browser. A relational database server stores whatever information the application requires. One of the easy languages could be used to handle requests between the web server and the database server is PHP by apache webserver.

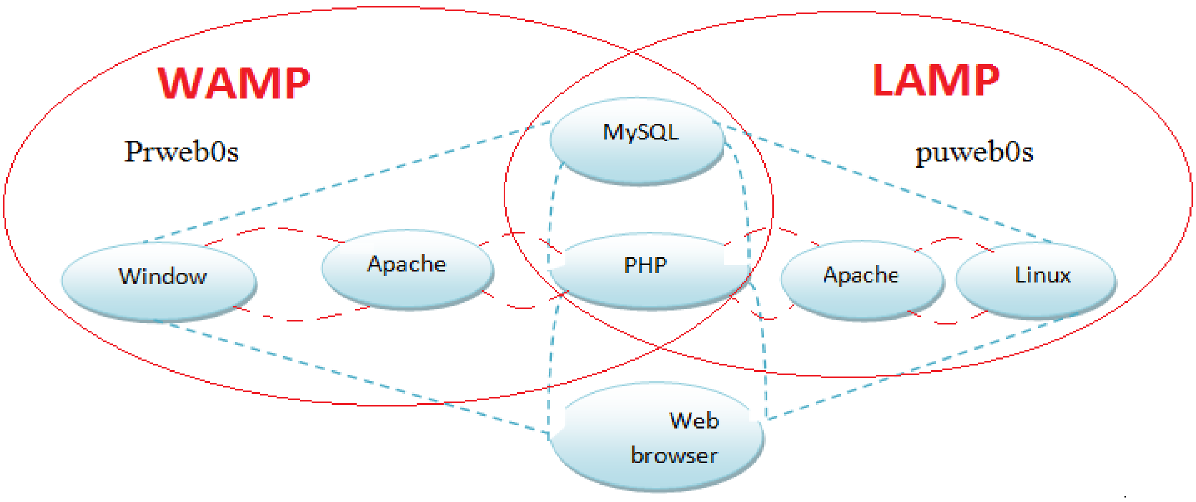


Fig (3.3) PrWebOS and PuWebOS over different OS

PHP is powerful and easy to use software (Hyper Pre- Processor -acronym originally person home page) which is a general scripting language that was designed for web development and to create dynamic web pages. PHP code is embedded into HTML source document and interpreted by web server. PHP is available as a processor for most modern web servers and used to handle requests between the Web server and the database server. Webserver is a system that delivers services to end users over the Internet and software used to facilitate HTTP (it is the key protocol used to transfer data on the web communication); it is also known as an (Internet server). All computers that host Web sites must have Web server programs. The primary function of a web server is to store, process and deliver web pages to clients. There are many types of web server such as IIS, Apache, etc. the Apache HTTP server also referred to Apache which is one of the most widely used web server software available. It is open-source and free that can be run on different operating systems like (Windows- Linux- Unix, Mac OS X and etc.).

In the proposed system XAMPP is working on intranet and the main components of the system are the web applications that create Virtual Desktop (VD) of user.

There are three different methods of WebOS related to the proposed system which are:

1) **Private Web Operating System (PrWebOS):-** The service is located on virtually separate infrastructure which is designed by us without need of Internet facility because it is only intranet with specific limitation. Where the user device is connect to router and can easily access to the web application server which distributes the signal to each Customer of income to the Web application that are linked by relying on IP address this method also called (offline network), Figure (3.4) shows the PrWebOS . There are several advantages of PrWebOS that summarized as:-

   A. Limited number of users allows only to whom inside the organization connected and access to WebOS.

   B. increasing of the efficiency, performance, quality and speed of communication as the communication is much better.

   C. The distance and numbers of users' limitation in PrWebOS help us to be more confidential and security of information of users in the back end- database system.

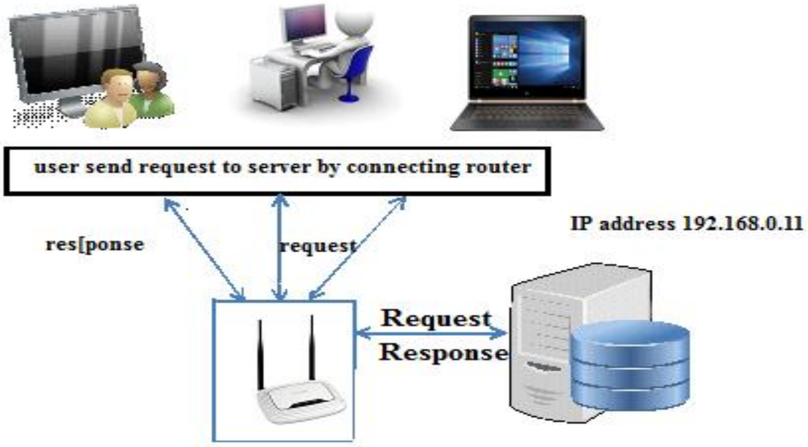


Fig (3.4) The private WebOS

2) **Public Web Operating System (PuWebOS):-** The users are access to WebOS with using PuWebOS method needs to the Internet facility with the shared infrastructure as a multiple consumers. Hence, security managements for such method is more necessary if compared to PrWebOS. Generally, when the numbers of users are increasing, the level of security will decrease. Figure (3.5) shows the PuWebOS.

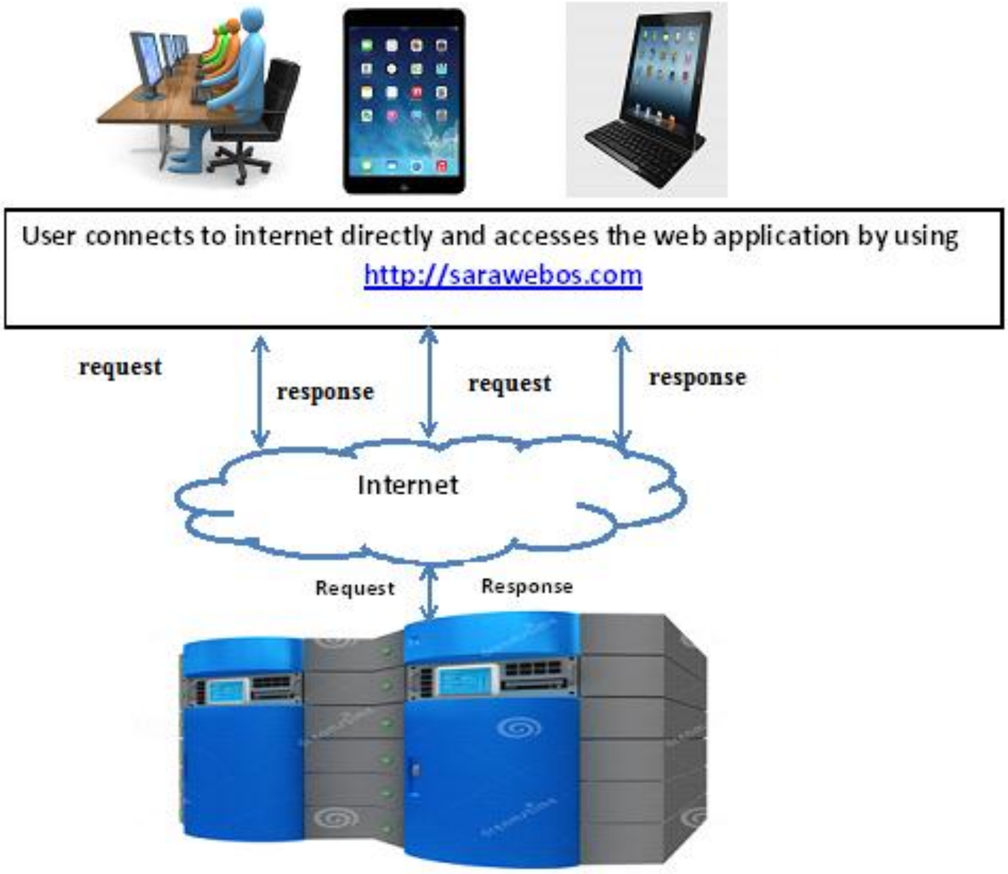

User connects to internet directly and accesses the web application by using
http://sarawebos.com
request
response
request
response
Internet
Request
Response

Fig (3.5) The public WebOS

Below are some disadvantages of PuWebOS comparing to PrWebOS:-

- Because the PuWebOS have more users, therefore the confidentiality and protection of information and data system is weak because of the intrusion that may occur from any third person can access to information easily.

- Also the distance to be non-limited and may be a large cross continents because it depends on the presence of the Internet in any place that allows user to login and access to application.Figure (3.6 A&B) shows the core work for proposed system that runs both types of WebOS namely (EyeOS & Lucid) over PuWebOS and PrWebOS with windowsOS and Linux TOS.

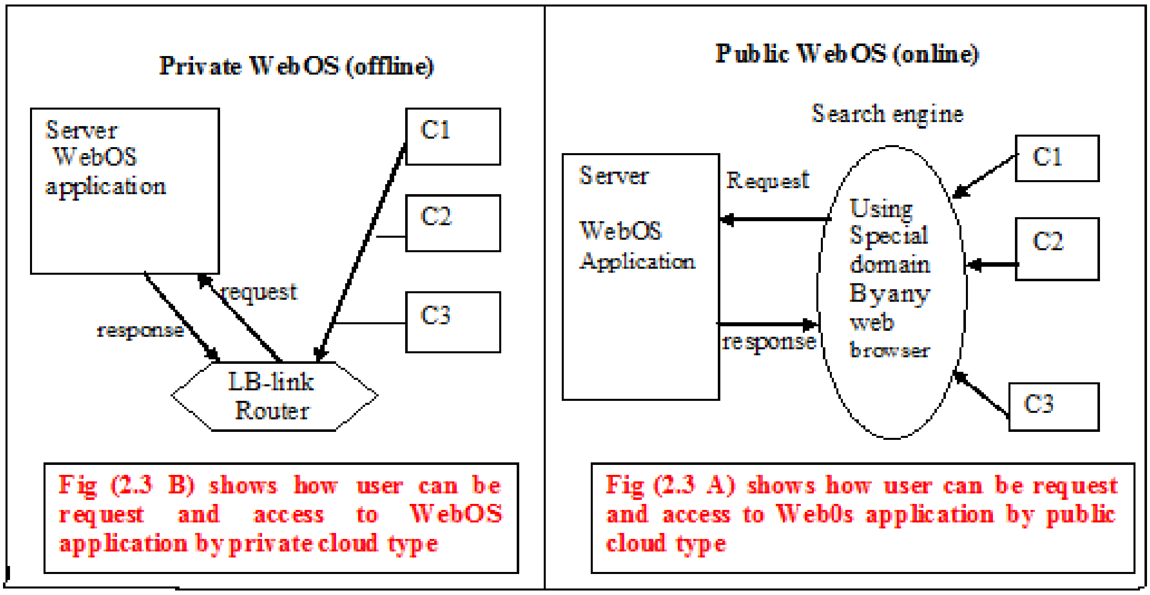

Private WebOS (offline)
Server
WebOS
application
C1
C2
C3
request
response
LB-link
Router
Fig (2.3 B) shows how user can be request and access to WebOS application by private cloud type
Public WebOS (online)
Search engine
Server
WebOS
Application
Request
response
Using Special domain Byany web browser
C1
C2
C3
Fig (2.3 A) shows how user can be request and access to Web0s application by public cloud type

Fig (3.6 A&B) The core work in proposed system with (private &public).

The environmental of WebOS shown in Fig (2.6 A&B) consists of two important types which are PuWebOS and PrWebOS. The two mentioned types of CC are exactly as same as to design WebOS. The first one is a PuWebOS and the second is the PrWebOS. In PuWebOS the information and application are hosted by third party and controlled by the Rent-As-Server URL provider .The Rent-As-Server domain name for our proposed system is http://sarawebos.com. The mostly used CC service model for PuWebOS is SaaS. SaaS is more popular for PuWebOS because when the WebOS proposed system will be completed and uploaded to the mentioned domain the only requirement is software. Thus SaaS is a most popular type used in PuWebOS.

However, the PuWebOS is exactly contrast to of PrWebOS in the use of SaaS, PaaS and Iaas which are used for designing the PrWebOS of the proposed system. The need of IaaS, PaaS and also customized SaaS [REES11] for their business in the private cloud but their business public Cloud is only utilize SaaS

3) **ChWebOS:** in the proposed system PrWebOS and PuWebOS are utilized in order to provide a better service. By blending of PrWebOS and PuWebOS, ChWebOS is the combination of WebOS methods.The proposed system is ChWebOS because client uses both WebOS types. Figure (3.7) shows blending of both PrWebOS and PuWebOS to create ChWebOS.

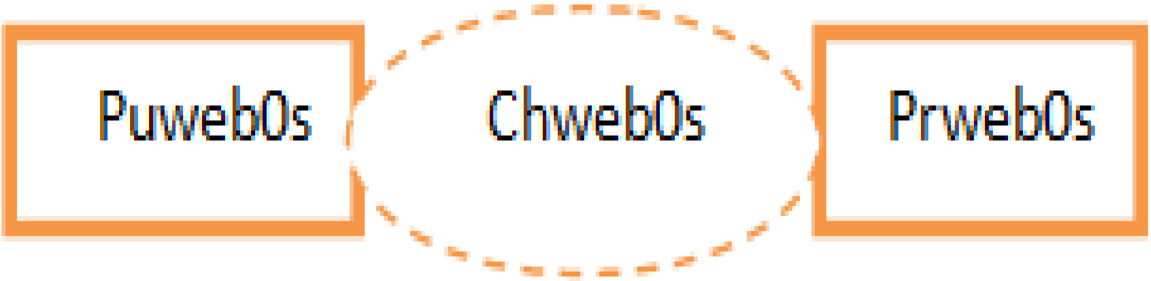


Fig (3.7) Blended of PrWebOS and PuWebOS create ChWebOS

3.3 System Requirements Interface

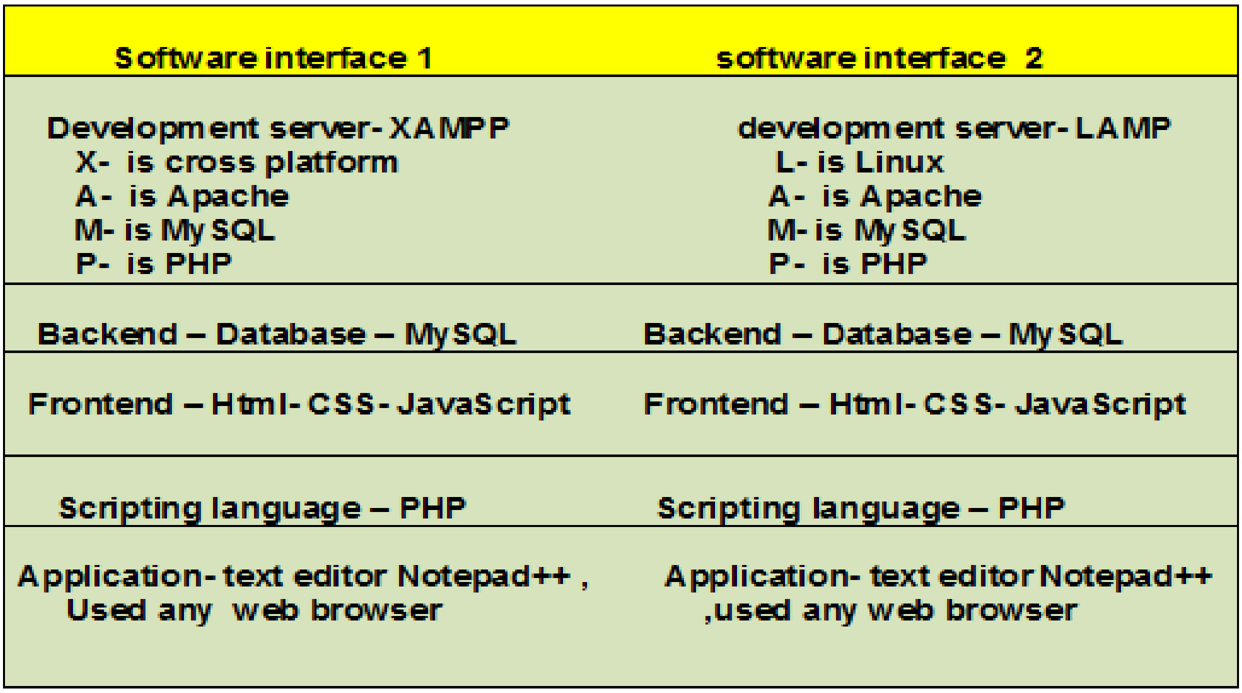
In order to design such WebOS, it is necessary to prepare the requirements of both of software and hardware interfaces.

A. Software Interfaces

To implement the WebOS proposed system, it is necessary to have several different types of computers hardware and software, the server side computer must be more powerful than the client side (due to the rule of networking and data communication).

The capacity of the server side computer must be very high in order to solve the problem of traffic network (Bottleneck). Normally, PuWebOS is with LAMP and its more powerful than our PCs that is connected to the server, also the server of the windows has a higher capacity net. The rate of traffic network (bottle neck) in both of them is roughly the same because the distance between the client and server is main factor. In general, Linux could be better because it has an open source, free and easy to download due to its compatibility with XAMPP. Linux unlike windows is the last that comes from a single company called (Microsoft). Linux does not belong to single company but varieties of companies produce Linux operating system under several names. Table (3.1) describes all software and applications that are used for designing proposed system.

Table (3.1) All software and applications that used for designing the proposed system

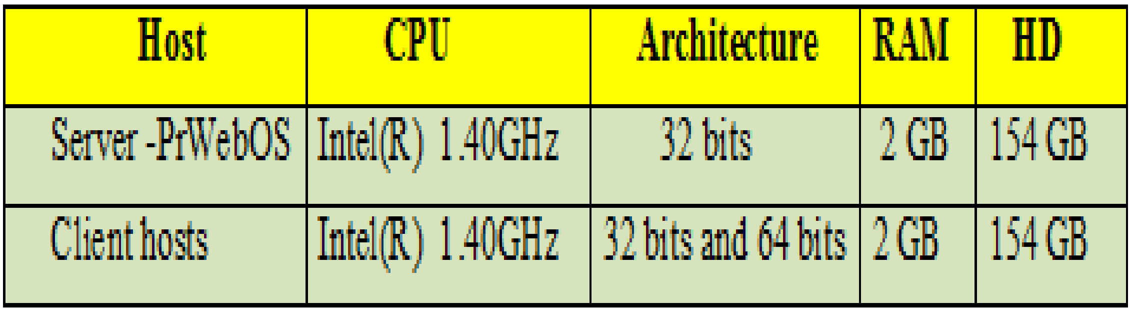


Software interface 1	software interface 2
Development server- XAMPP X- is cross platform A- is Apache M- is My SQL P- is PHP	development server- LAMP L- is Linux A- is Apache M- is My SQL P- is PHP
Backend – Database – MySQL	Backend – Database – My SQL
Frontend – Html- CSS- JavaScript	Frontend – Html- CSS- JavaScript
Scripting language – PHP	Scripting language – PHP
Application- text editor Notepad++ , Used any web browser	Application- text editor Notepad++ ,used any web browser

B. Hardware Interfaces

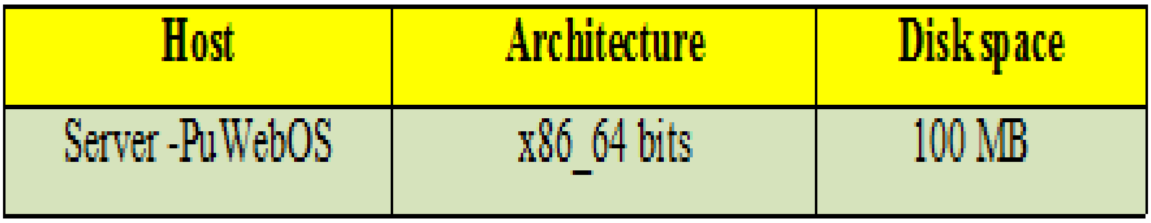
The hardware features of depended equipment include: (hosts and router) in this system are shown in tables (3.2A&B) and (3.3). In order to provide more practical applications and to be close to the real situation, the system has been implemented using hosts with different features and abilities. Table (3.2A&B) shows all hardware equipment that features for the proposed system machinery. The only explanation need for this table is the small size of memory space of the domain. The domain is just used for testing which 100MB was enough. Its small size for such application in reality but for implementing is enough. Also, Table (3.3) shows the features of router depended in the proposed system.

Table (3.2 A) All hardware equipment that features for the proposed system Machinery (PrWebOS)



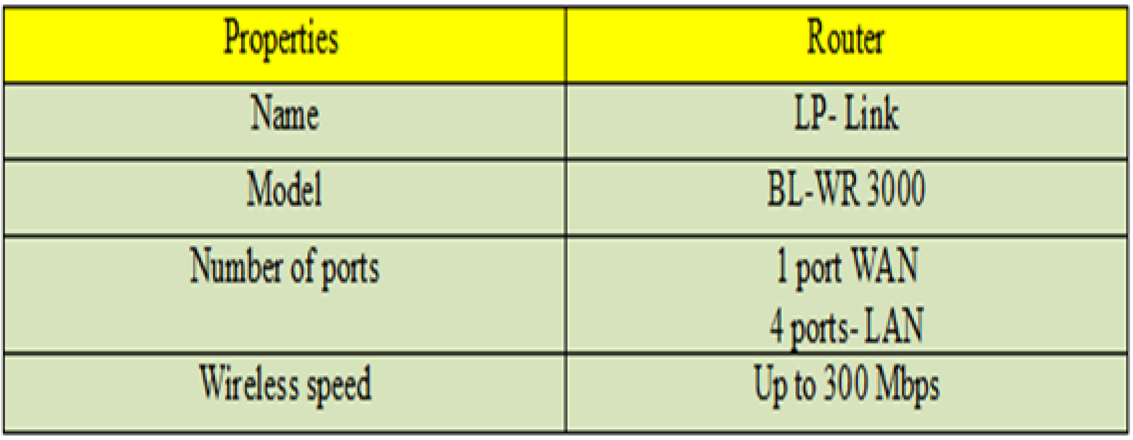
Host	CPU	Architecture	RAM	HD
Server -PrWebOS	Intel(R) 1.40GHz	32 bits	2 GB	154 GB
Client hosts	Intel(R) 1.40GHz	32 bits and 64 bits	2 GB	154 GB

Table (3.2 B) All hardware equipment that features for the proposed system machinery (PrWebOS)



Host	Architecture	Disk space
Server -PuWebOS	x86_64 bits	100 MB

Table (3.3) shows the features of router depended in the proposed system



Properties	Router
Name	LP- Link
Model	BL-WR 3000
Number of ports	1 port WAN 4 ports- LAN
Wireless speed	Up to 300 Mbps

3.4 System Functionality

There are several steps to design the EyeOS and Lucid:

1. Install Notepad + + in the server side of the proposed system to typing Html and PHP code.

   Rationale: the use of notepad ++ for saved EyeOS and Lucid desktop codes in the localhost of the computer server.

2. Installed of compatible software X cross platform, Apache, MySQL, PHP and Perl that called (XAMPP 1.8.2-6) as a web technology type.

   Rationale: (X) is the cross platform and supports all types of operating system. Apache is a web server and an excellent middleware in-between frontend as Web browsers- SaaS and backend as MySQL -PaaS. PHP -IaaS and SaaS are making any website dynamic or it is a spirit of the WebOS system types due to all changeability of data in our WebOS.

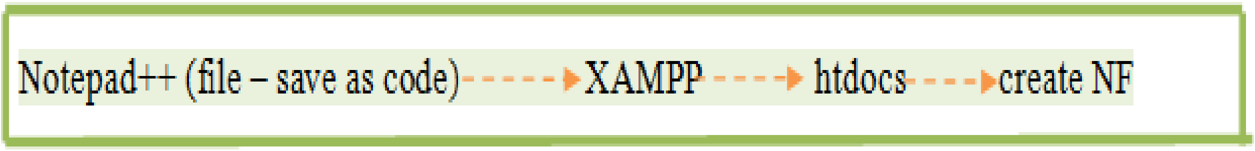
3. Typing codes in Notepad+ +and save it in localhost of the computer server

   Rationale: the save codes in notepad++ once, it will be a server side for the PrWebOS and another time it becomes a client for the PuWebOS.

4. PrWebOS is an intranet and PuWebOS is internet.

   Rationale: the Uniform Resource Locater (URL) of the PuWebOS is: http://sarawebos.com . But PrWebOS IP address is http://192.168.0.11 .

In PuWebOS the administrator should upload all notepad^{++} files and folders by File transfer protocol (FTP) namely FileZilla to http://sarawebos.com But, in PrWebOS the administrator only save at localhost / htdocs /XAMPP. Figure (3.8) shows the way of saving in XAMPP localhost.



Notepad++ (file – save as code) -----> XAMPP ----> htdocs ---> create NF

Fig (3.8) The way of saving in XAMPP localhost

5. PrWebOS users' domain is http://192.168.0.11, but PuWebOS domain is http://www.sarawebos.com.

   Rationale: after clicking to both URL types, the users can see user interface

6. Registered WebOS users for EyeOS.

   Rationale: unregistered consumers to WebOS (i.e EyeOS) must click to the hyperlinked **new user** alphabetises. After that the users must be filling both(username and password) fields and click **create account**. The Registered of consumer can easily enter a correct username and password in order to login to WebOS. It is impossible for users duplicate username due to the WebOS is intelligent system. Hence, the system never let the same username to create account (duplicated prevent).But a warning message of duplicated username by the proposed system is username already exists. The conversational description for the new customer registration for EyeOS is given below:

Enter *your information*
Consumer Username
Consumer Password
Check *Completeness and Integrity*
Register

Figure (3.9) shows the flow chart of registration clients for WebOS (EyeOS).

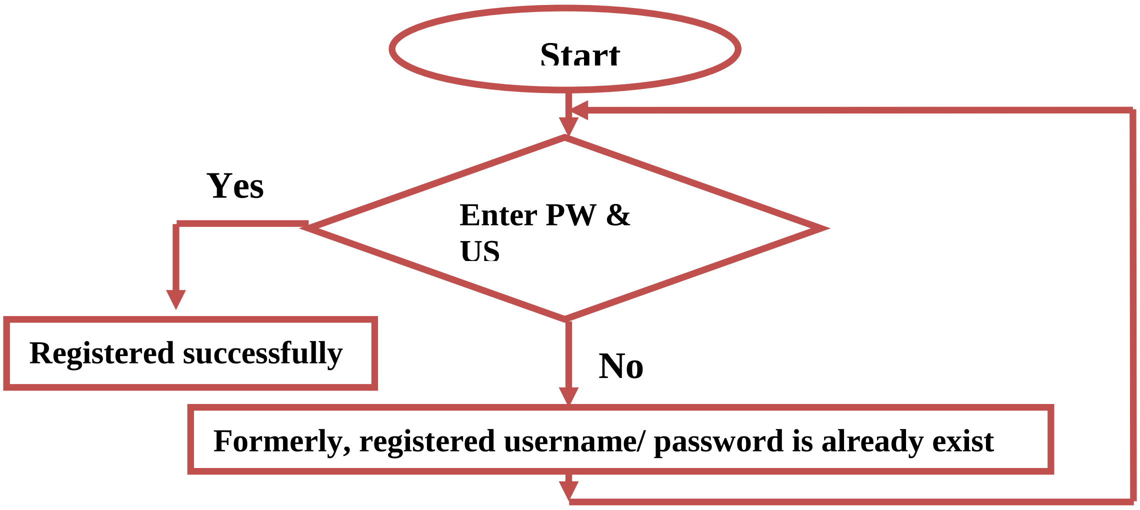


Fig (3.9) The flow chart of registration clients in WebOS (EyeOS)

7. Login: WebOS users to EyeOS.

Rationale: any system (i.e. EyeOS) is normally requiring the clients (username and password). In order to pass a login form, it's better to register or users have correct username and password. To registered to the system with the holding a correct (username and password) is simply access to the EyeOS. The conversational description for new customer login for EyeOS is given below.

***Enter** your information*
Consumer Username,
Consumer Password
***Check** the availability of such account*
Login

Figure (3.10) shows the flow chart of login clients for EyeOS WebOS

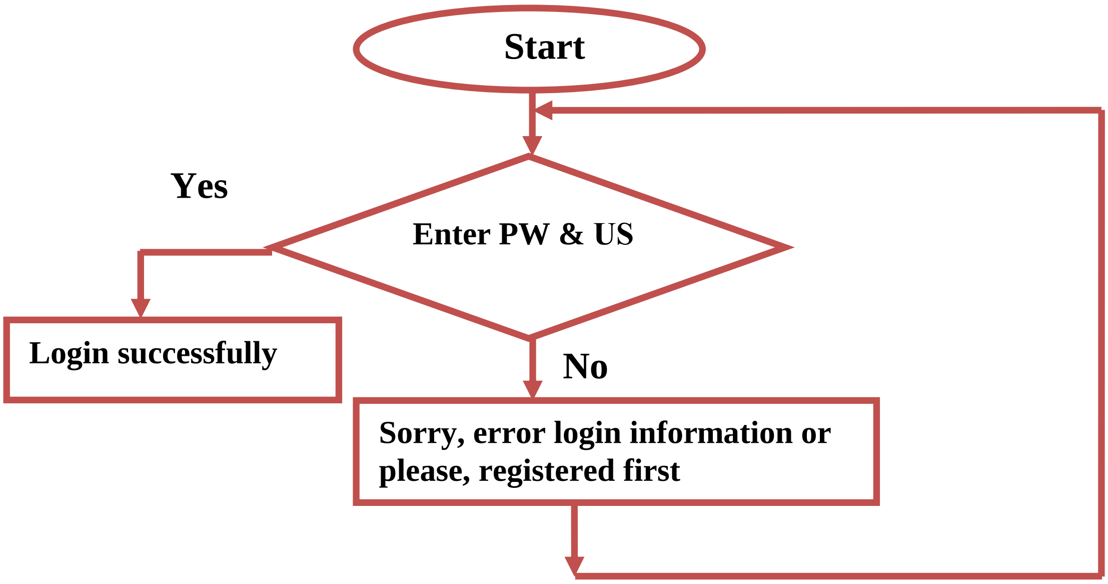


Fig (3.10) the flow chart of login client in EyeOS system

8. Entered: the consumer entered to WebOS

   Rationale: the consumer with a correct username and password simply has an access to WebOS and free to use all available applications. The examples of EyeOS file manager are: Desktop, document, image, music, video, games and etc. Nevertheless, the steps one to five is exactly suitable for both the EyeOS and lucid. The similarity of the steps for (EyeOS and lucid) are steps of one to five but the steps six to eight are dissimilar. For that reason, should be rewrite the steps six, seven and eight: as following.

9. Registered: WebOS users for lucid.

   Rationale: unregistered consumer to our WebOS proposed system must click to hyperlinked register alphabetises. After that the users must fill the form of (Username, email, password a confirm password fields and clicks **submit**. The registered consumer can easily enter a correct username and password. In order to login to the lucid desktop, It is impossible for lucid users to duplicate the username due to the WebOS intelligent system. Hence, the system never let the same username to create account (duplicated prevent). But, a warning message of duplicated user's names by the proposed system the username has already existed. Figure (3.11) shows the flowchart of the registration clients for lucid. The conversational description of registered customer for lucid WebOS is given below.

***Enter** your information*
Consumer Username
Consumer Password
Consumer email
Confirm password
Submit

Figure (3.11) shows the flow chart of registration clients for Lucid WebOS

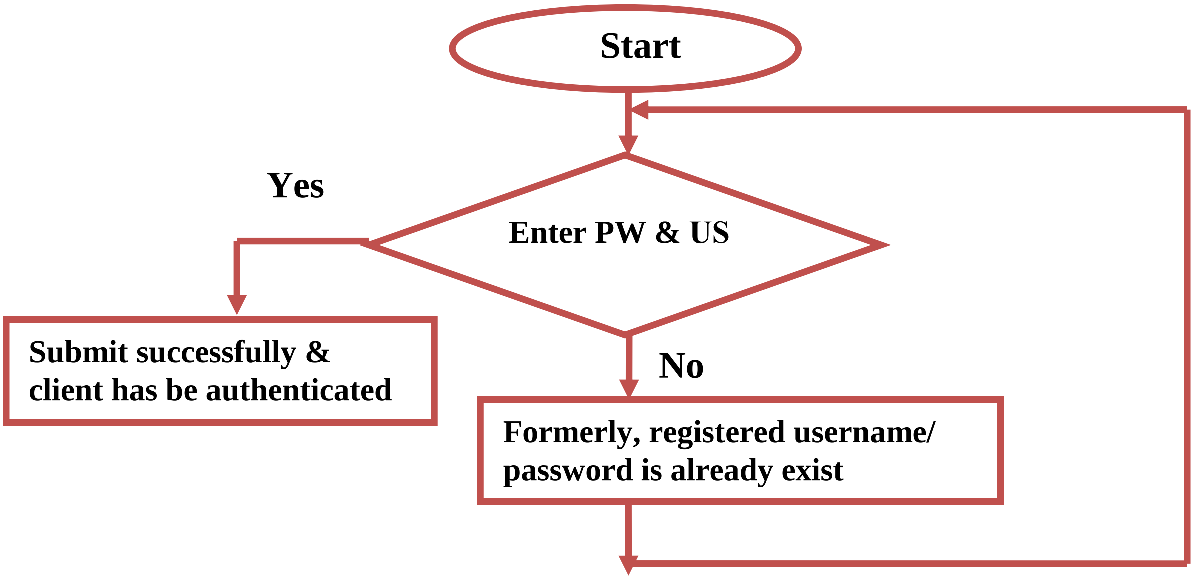

Start
Yes
Enter PW & US
Submit successfully & client has be authenticated
No
Formerly, registered username/ password is already exist

Fig (3.11) The flow chart of registration clients in WebOS (Lucid).

10.Login: WebOS users to lucid.

Rationale: any system (i.e. lucid) is normally requiring client's username Password, email, confirm password. In order to pass a login form, it is better to register and users must have a correct username and password. The registered to the system with holding a correct username, password, email and confirm password is simply access to the lucid system. The flow chart of login of Lucid is like EyeOS that mentioned in figure (3.10). The conversational description for the new customer login for lucid is given below.

***Enter** your information*
Consumer Username,
Consumer Password
Login

11. Entered: the consumer entered to lucid.

Rationale: the consumer with correct username and password simply has an access to the WebOS and free to use all available applications. The examples of lucid file manager are: accessories, games, graphics and etc.

3.5 WebOS Categories

There are lots of categories of WebOS; in this thesis the emphasis is on two of them as showed in figure (3.2).

3.5.1 EyeOS

It is a web based system and an open source across platform of Linux which is the whole system placed in server side for both PrWebOS and PuWebOS. EyeOS is designed by Apache server, PHP and MySQL database server over the cross (X) platform namely is (XAMPP) and run over windows and Linux.

The considered of WebOS because it is more popular that allows the users to have access through portable devices such that (smart phone and computers) and accessible using web browser to run wide variety of applications like (word processor, address book, contacts, PDF reader, open files and etc.). It's let users to deal with web applications for (listen to music or video, store data, playing games and etc. Figure (3.12) shows the accessing of EyeOS by using any browsers.

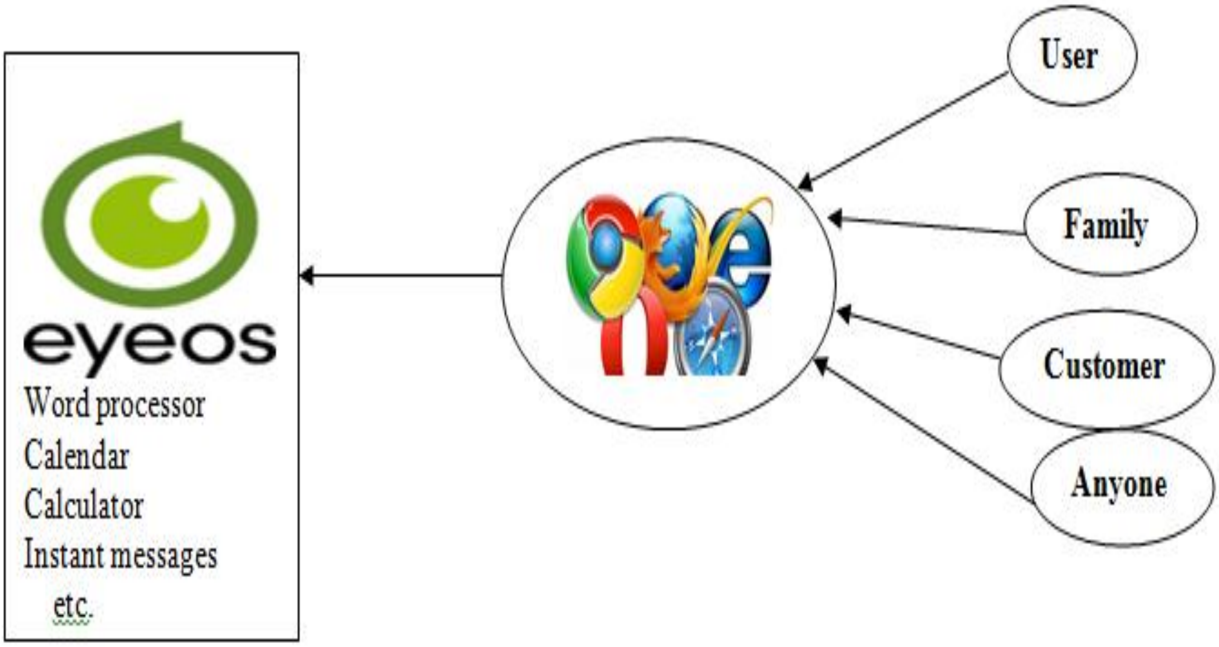


Fig (3.12) The user accessing to EyeOS

After writing all codes in notepad^{++} by PHP and saved in a special folder that named EyeOS. The availability of XAMPP is one of the conditions to work dynamically and run in the server side. The figure (3.13) shows all steps to run in server side:

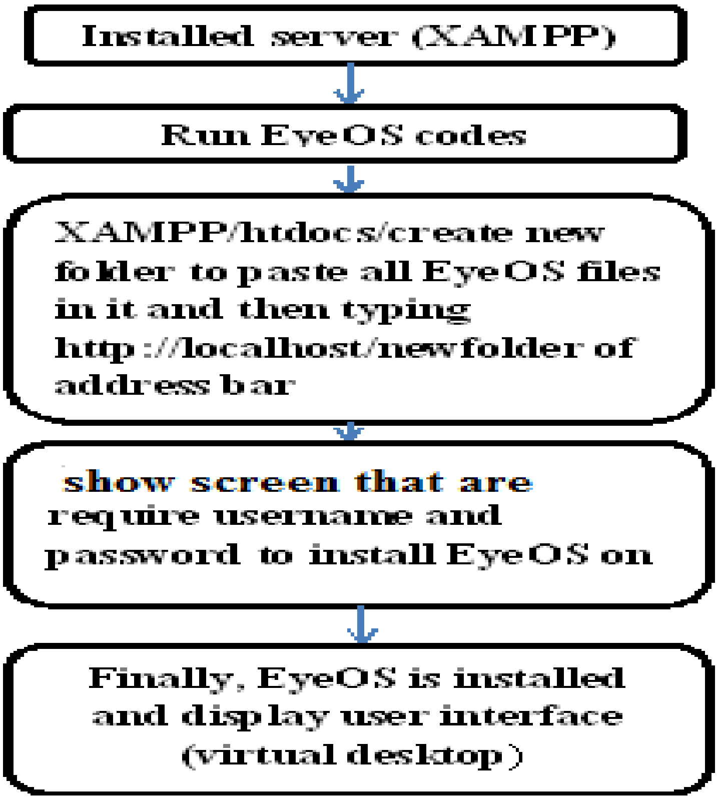

Installed server (XAMPP)
Run EyeOS codes
XAMPP/htdocs/create new folder to paste all EyeOS files in it and then typing http://localhost/newfolder of address bar
show screen that are require username and password to install EyeOS on
Finally, EyeOS is installed and display user interface (virtual desktop)

Fig (3.13) The steps of running EyeOS on server side

Finally, it is clear with more details about how to implement of EyeOS/ lucid desktop (same mechanism) and user friendly.

3.5.2 Lucid Desktop

Basically the lucid desktop runs on the server side a like EyeOS. It allows consumer to enter the WebOS lucid applications in an anywhere anytime to execute request such as (listen to music, browse photo, edit document and save files). The user only needs to use any web browser and internet facility to communicate in the proposed system PuWebOS. But for PrWebOS using of IP address 192.168.0.11. Lucid system applications are (Rss feed reader, games, calculator and etc.).

The installations of PHP and MySQL with apache web server are an important factor on server side for design lucid. The figure (3.14) shows the accessing of lucid by any browser in the client computer.

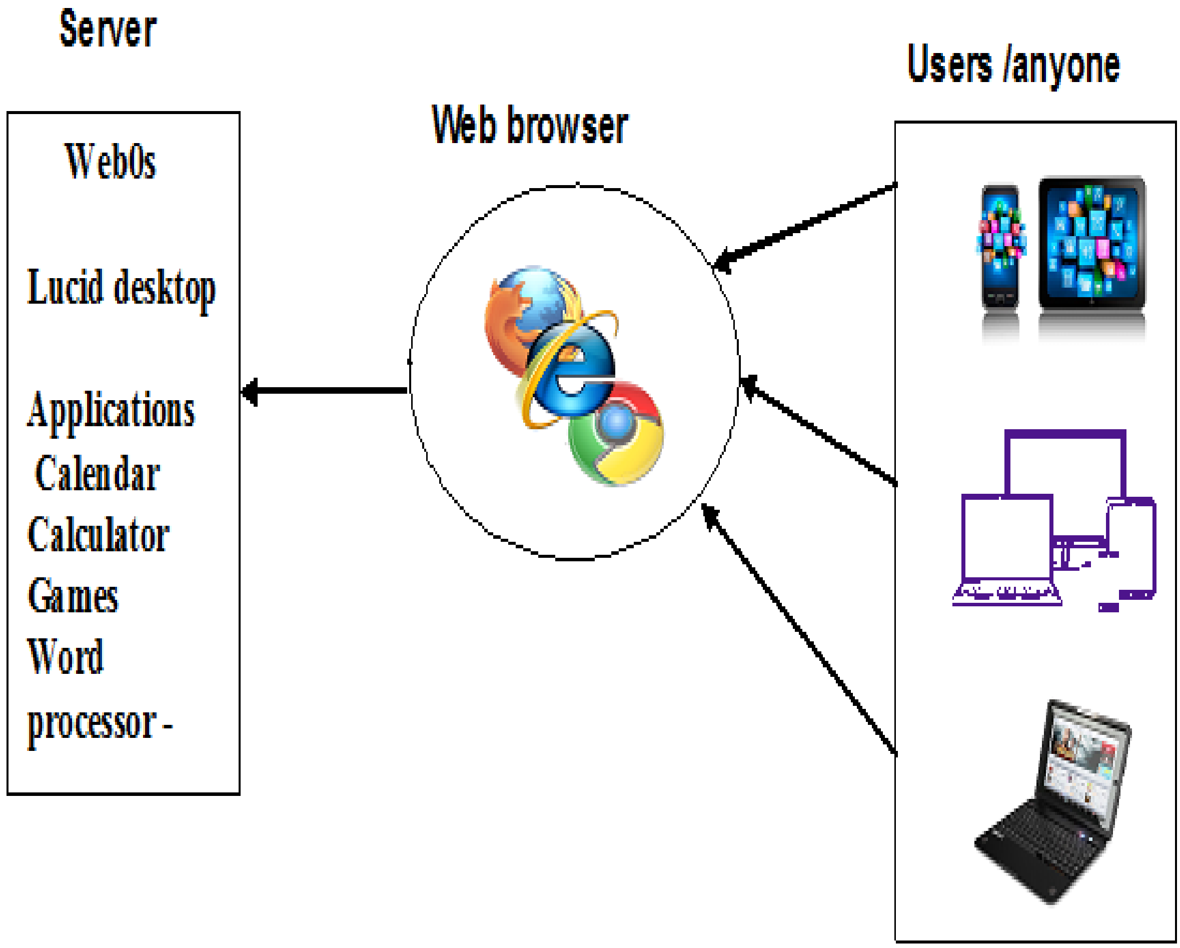


Fig (3.14) The user accessing to lucid desktop

After writing codes in notepad++ by PHP and saved in a special folder .The availability of XAMPP is one of the condition to work dynamically which run in the server side. The figure (3.15) shows all steps to run in server side. As mentioned before the steps are as same as to EyeOS.

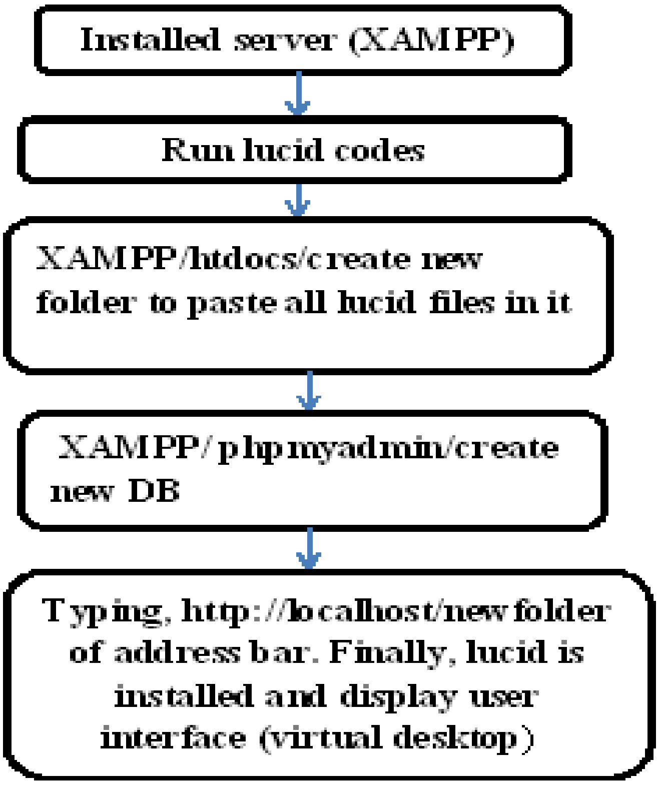

Installed server (XAMPP)
Run lucid codes
XAMPP/htdocs/create new folder to paste all lucid files in it
XAMPP/phpmyadmin/create new DB
Typing, http://localhost/new folder of address bar. Finally, lucid is installed and display user interface (virtual desktop)

Fig (3.15) The steps of running lucid on server side

The implementation access of lucid desktop (same mechanism to EyeOS)

Recommendation: The main recommendations of the proposed system are issued for both EyeOs and Lucid using two mechanisms. Each mechanism has its own user's interface and shows different applications accuracy for clients. The proposed system gives a chance to select either lucid or EyeOS. The test is very accurate because of the one computer can be tested each time. The capacity and speed of the client or server that test are constant. Since the capacity and speed are unchanged the result will be accurate.

Chapter Four
The Implementation of Proposed System

4.1 Introduction

This chapter illustrates the established WebOS of the proposed system to communicate between clients and server (C/S) based on PrWebOS or PuWebOS over two different types of operating system. PrWebOS is an intranet designed by wireless LP-Link router for distance limit of hundred meters with the domain http://192.168.0.11, but PuWebOS is an internet designed with hire domain of http:/sarawebos.com. In the proposed system the server used of two OS "windows and Linux" and clients can use any type of browser to login to WebOS applications. Windows for PrWebOS, but Linux for PuWebOS were built by using PHP and MySQL. A lot of hyperlinks are used to facilitate for users' navigation for recommendation purposes. In an index page there are two hyperlinks to select one of them either EyeOS or lucid from the client.

Besides WebOS, the second part of the system links between these WebOS and the database. The database stores the essential information about WebOS applications, users, administrator(s) and users' request. The handling of data transactions from database to WebOS is using by (MySQL) technology. The established WebOS for the two types of EyeOS and lucid that tests them by using two methods of network (PrWebOS and PuWebOS).

4.2 The Testing Media

The testing media of the WebOS is done by two methods, the first one is net-centric; internet (i.e. PuWebOS) and intranet (i.e. PrWebOS). However, the second one is Non-net-centric (i.e. TOS). The net-centric is electronically and the Non- net centric is fully traditional.

First part (PrWebOS)

1- Create a small wireless network -intranet which established for the purpose of testing the PrWebOS of the proposed system. The topology of the established PrWebOS connects several (clients) to server through a (LP- link router). Figure (4.1) shows the topology of the established PrWebOS (small wireless network).

There are several steps to establish the small network:-

A- Installing XAMPP server on windows – 7 OS.

B- Prepare (EyeOS and lucid) WebOS on server and run the codes.

C- Connect users to the server through the mentioned router and providing them IP address http://192.168.0.11 .

D- It is absolutely local within a company use up to hundred meters or a bit more.

Installing XAMPP web technology on the top windows OS in server side that include: PHP, Apache and MYSQL with windows (WAMP).

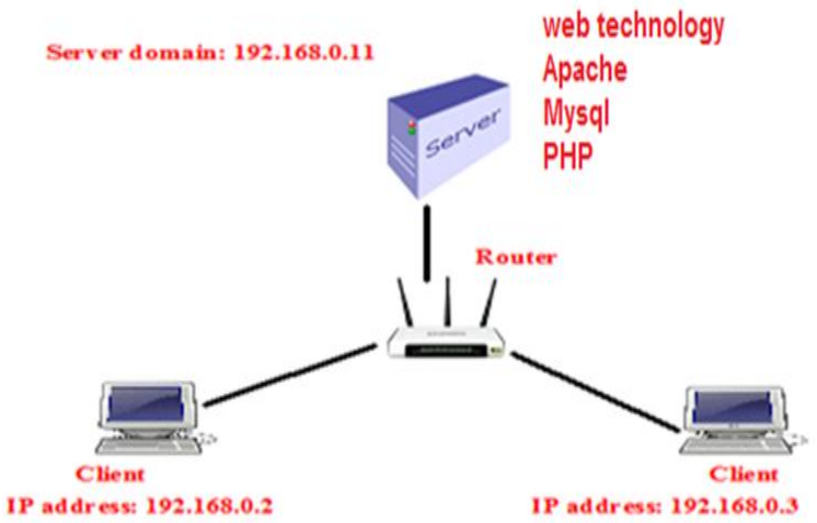


Fig (4.1) The topology of the established PrWebOS (intranet)

Second part (PuWebOS)

2- Built wide area network that has internet facilities and allows users to access server (web applications):

A) Using special domain http://.sarawebos.com that hired for the purpose system.

B) The hired server operating system is Linux, using the web technology (Apache , database MYSQL and middleware PHP) that becomes (LAMP).

C) EyeOS and lucid desktop are designed on any computer with internet facility could be TOS. After running and testing both of EyeOS and lucid codes must uploads to domain by FTP.

D) It is absolutely global.

Figure (4.2) shows the topology of the established PuWebOS (wide area network).

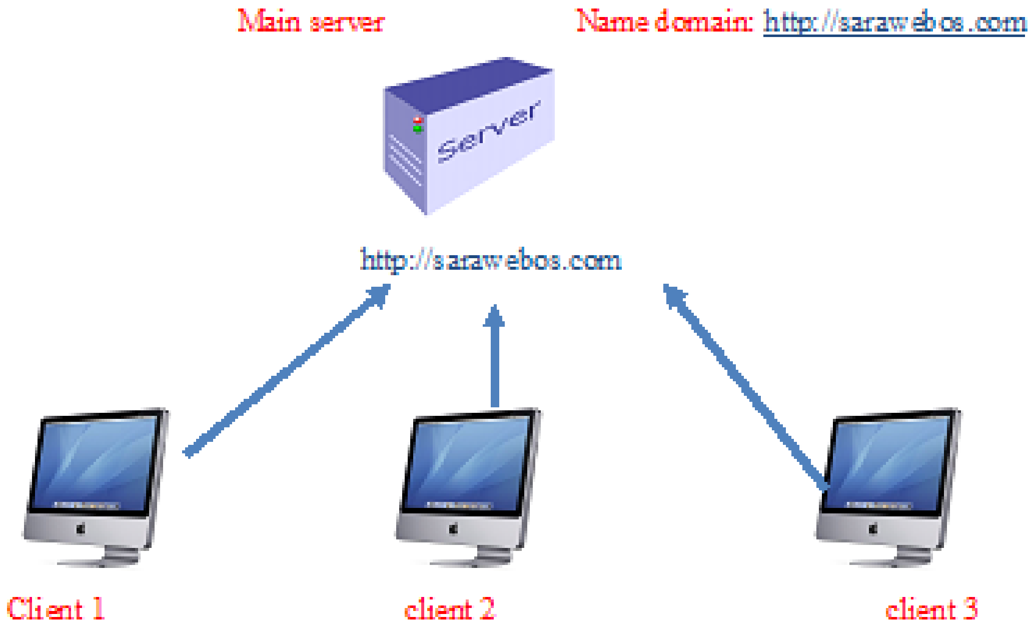


Fig (4.2) the topology of the established PuWebOS (internet)

4.3 The Established Prototype of the Proposed System

In this thesis, the established WebOS is supported by some models and methods that are designed and implemented. The features of the models are reviewed in chapter two, while the functional descriptions of the system modules are given in chapter three. In this chapter, the aspects of the established prototype are described, starting from design the virtual interface, applications, system administrator, and ending with structure of the system database. In our proposed system, two types of WebOS which are: EyeOS and Lucid. The wide area of network-internet (PuWebOS) is by using the domain and allows client accessibility to system in any part of the world.

4.3.1 EyeOS interface and webpages

This section describes the main EyeOS index webpage. The main webpage will be opened by entering the proposed system URL in the web browser address bar http://locaohost/EyeOS or http://sarawebos.com/EyeOS. The virtual desktop of EyeOS appeared with all entitled application in an EyeOS. Figure (4.3) shows the virtual desktop of EyeOS (the first webpage appeared).

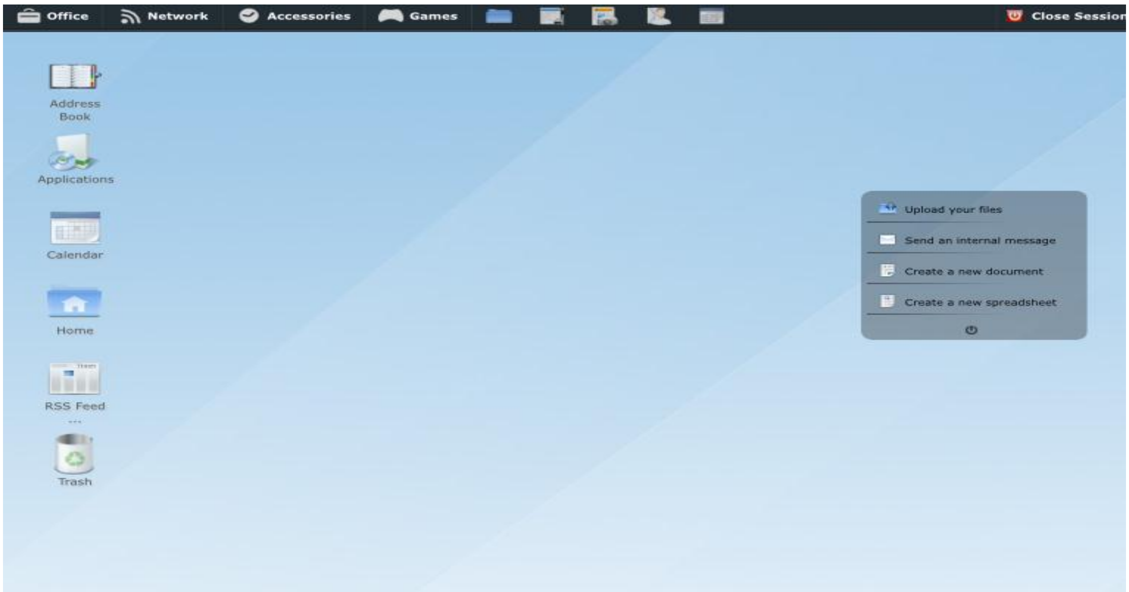


Fig (4.3) The standard webpage of EyeOS

In order to access the EyeOS-virtual Desktop that mentioned in figure (4.3), there are many steps should be followed by the user to be familiar to the system which are:-

A. client Registration

A new user cannot access to EyeOS unless has been registered. The new user must register in database to login the system. The unregistered client cannot get access to the mentioned system. Any WebOS (i.e. EyeOS) there is should be a registration form allows client to register information in database in order to become a member; otherwise the unregistered client cannot get access to the mentioned system. The registration form consists of the following information to be filled in from that related to the registration:

- **New user**: is the name that is used by the user to login of the system.
- **Password**: it is used for user's authentication to prove the identity access to the system which includes (alphabetic, characters and numbers).

Figure (4.4) The user registration page of EyeOS

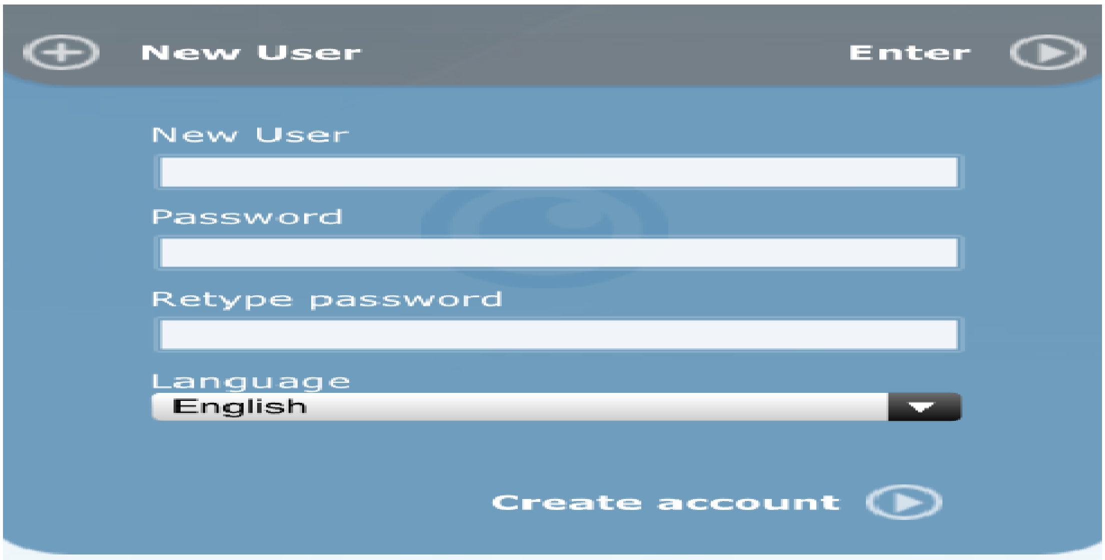
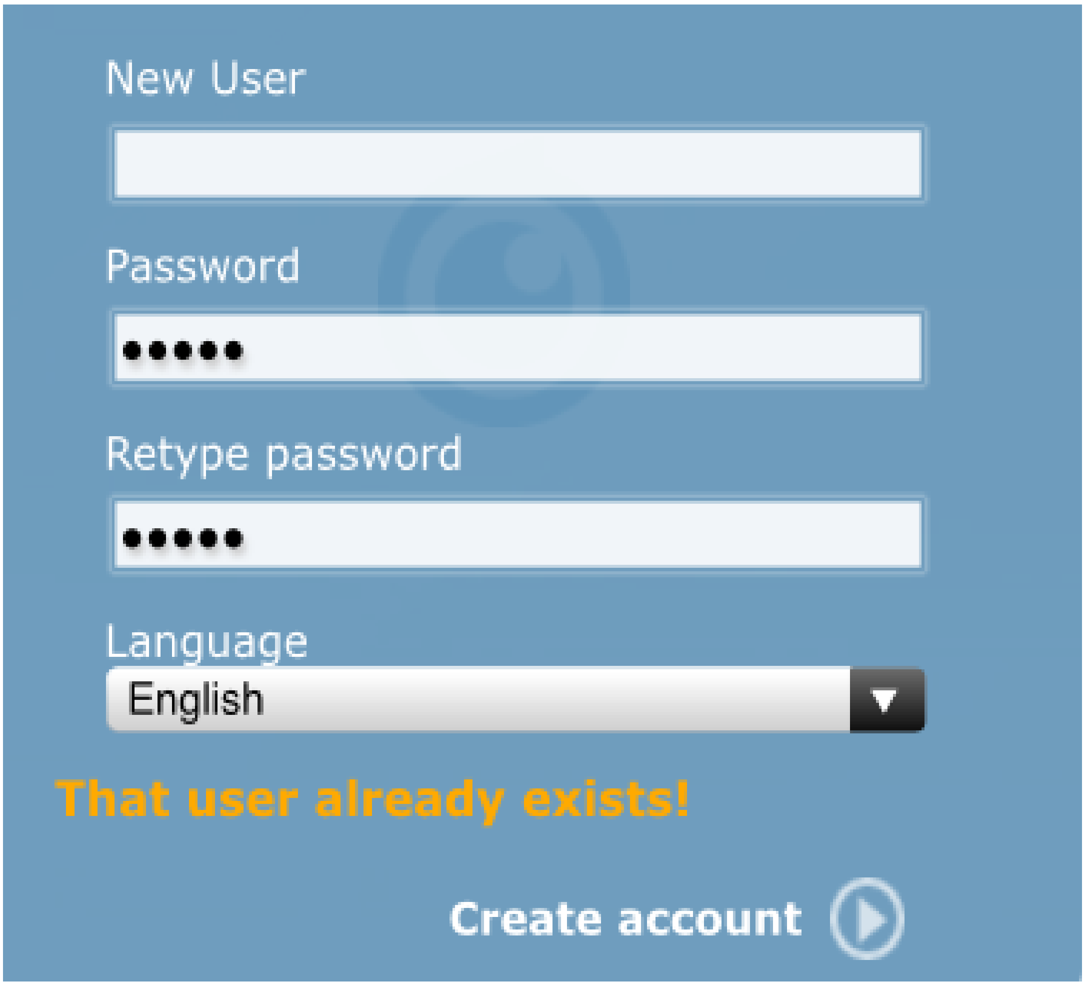


Fig (4.4) the registration pages of EyeOS

The system checks the validity of the filled data using database contents. While a user filled in and complete all registration information (username, password). If the user name is not pre-registered or duplicated in the database a new record will be created and stored in the database. Finally, the entered new username must not be matching with any username in the database, otherwise it is duplicated. It is impossible the proposed system to register new user if it is available in a database. The duplicated of new user name in the database will give a warning message to the user and must change it.

Figure (4.5) shows the warning message of username duplication.



New User

Password

●●●●●

Retype password

●●●●●

Language

English ▼

That user already exists!

Fig (4.5) The warning message of duplication.

B. Users Login

Logging-in is a basic step that comes after registration. The registered user can access to EyeOS system by entering correct (username and password). Figure (4.6) shows the user login of EyeOS.

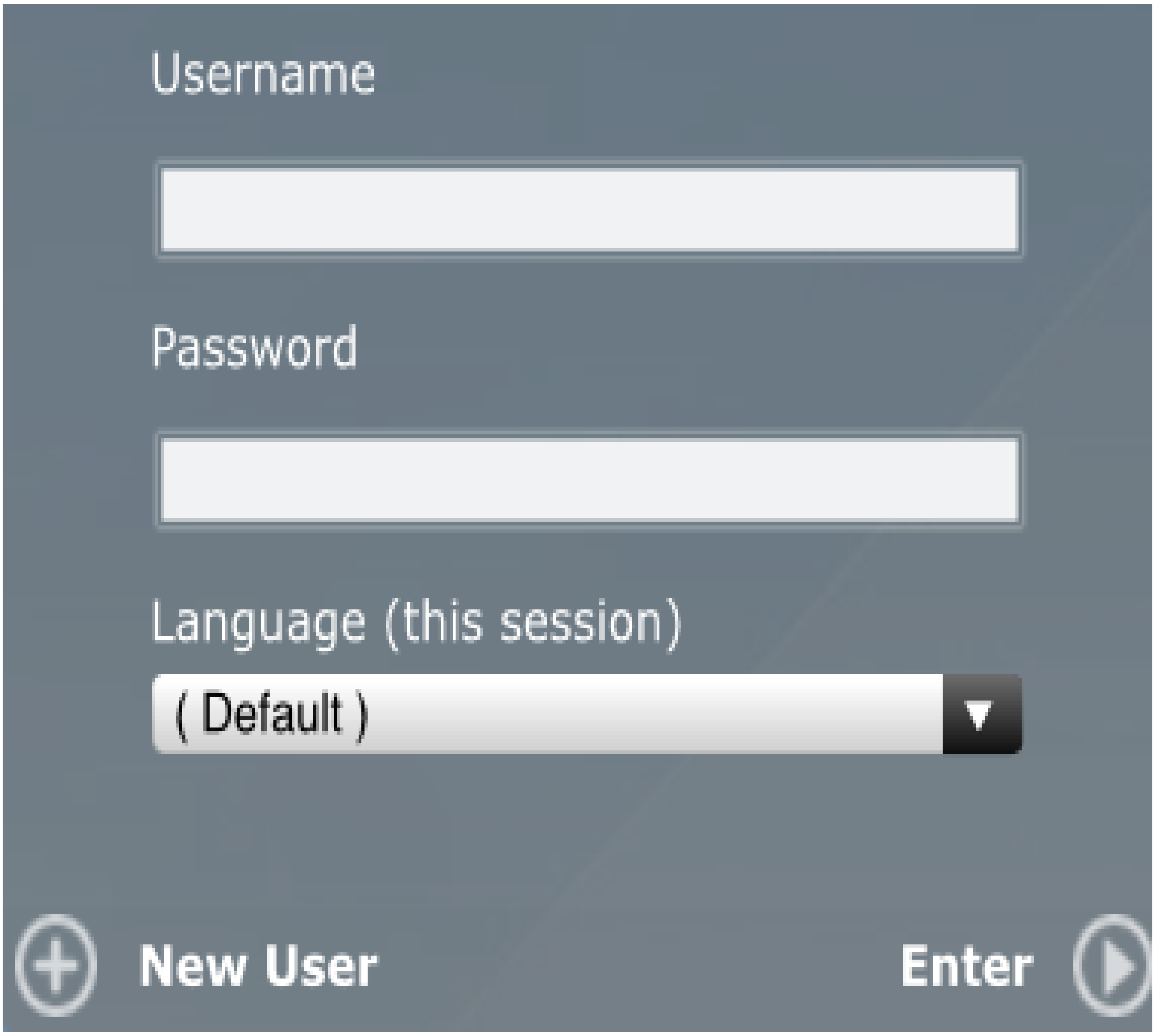


Fig (4.6) The login page of EyeOS

When a user enter his own username and password, the PHP script will compare them with those in database, and if they are correct, the script reads the user details such (username, password) and then log in to the system.

Finally, the page is ready to use from the user and can use the applications (such as address book, calendar, upload files, send internal massages, create new document and etc.) Figure (4.7) shows the next web-page- screen that appeared user logging correctly that included applications.

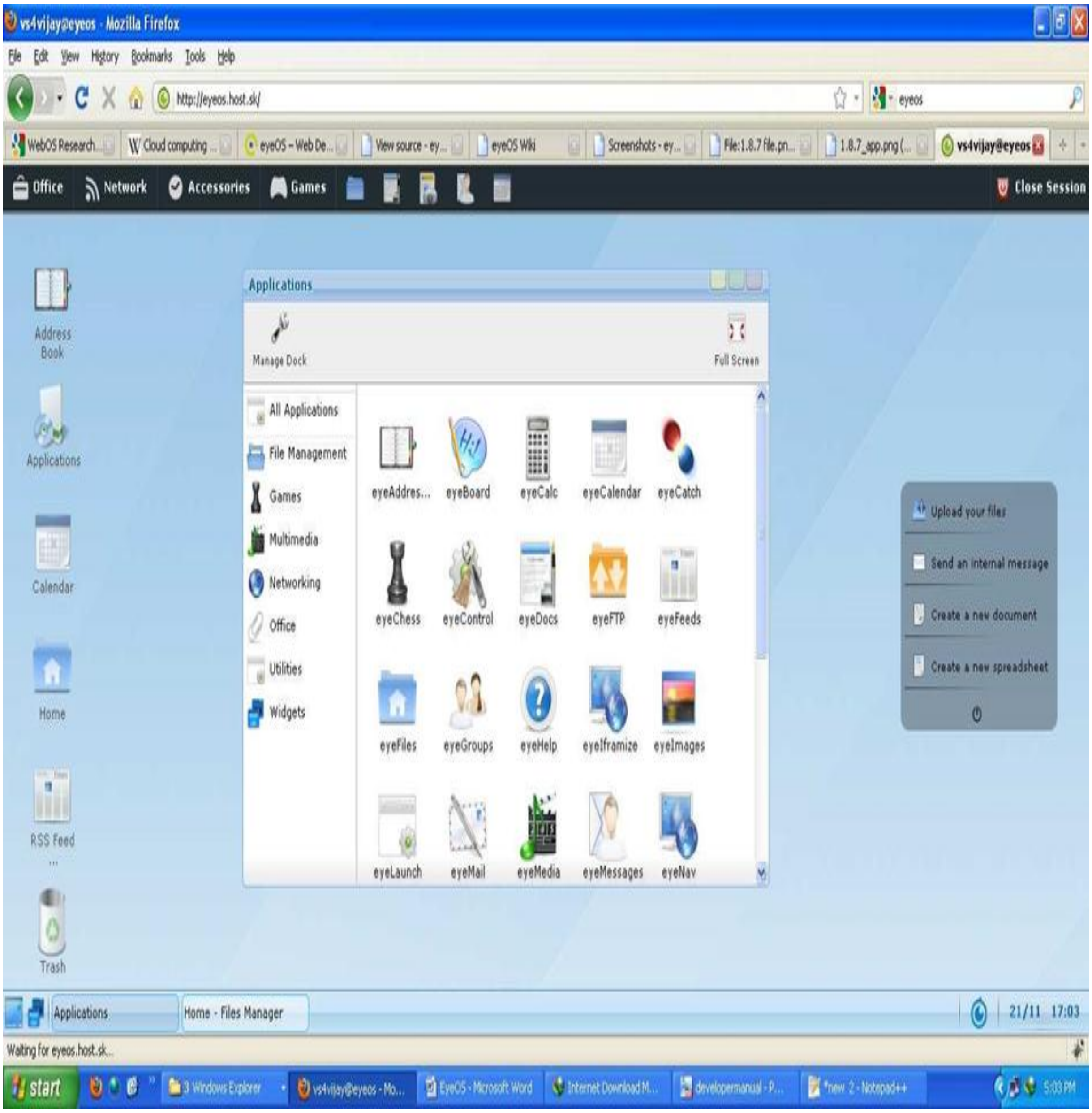


Fig (4.7) The next page that appeared after login page of EyeOS

4.3.2 Lucid interface and web pages

This section describes the main lucid index webpage. The main webpage will be opened by entering the proposed system URL in the web browser address bar. The virtual desktop of Lucid appeared with all entitled application. Figure (4.8) shows the main webpage of lucid.

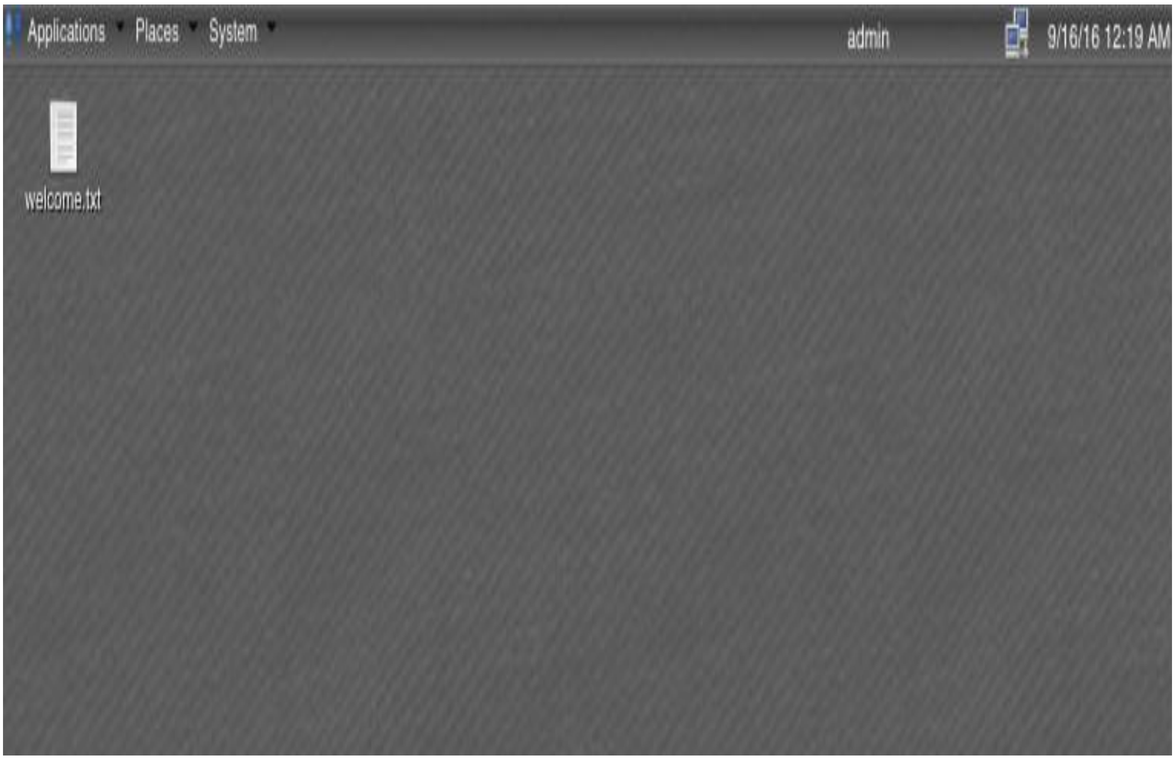


Fig (4.8) the standard webpage of lucid

In order to use the Lucid -virtual Desktop, there are many steps that should be followed by the user to become familiar to the system which is:-

A. client Registration

The new user cannot access to WebOS (Lucid) unless has been registered. The new users must register to a database to login to lucid system. The unregistered client cannot get access to mentioned system. Any WebOS like (lucid), there should be a registration form that allows client to register information in the database, in order to become a member; otherwise the unregistered client cannot get access to mentioned system. The registration form consists of the following information to be filled in from that related to the registration. The registration form consists of the following buttons that can be filled in with information from the user.

- **Username**: it is the especial name that used by the user to login system.
- **Email**: it has a great role in WebOS that facilitates the privacy for user's communication.
- **Password**: it's a main part that used to get authorization for access system.
- **Confirm password:** re-typing of the password is to certify that the user double check of the password and must be match each other.

In the lucid desktop system, the new user wants to register to the proposed system by only filling the mentioned form for the purpose of registration. After a click on "register button” the contents that filled up by the user will be stored in database, and the registered user will become capable to access the system. Figure (4.9) shows the user registration page of mentioned the system. The "reset password" button is another choice in our proposed system in case of any mistake. Hence, any mistake through user filled up password. In another case, when users have registered formerly and again registering by filled up matching information of username and password. The duplicated user name will be warning by message the message **“the user already exists"**. Figure

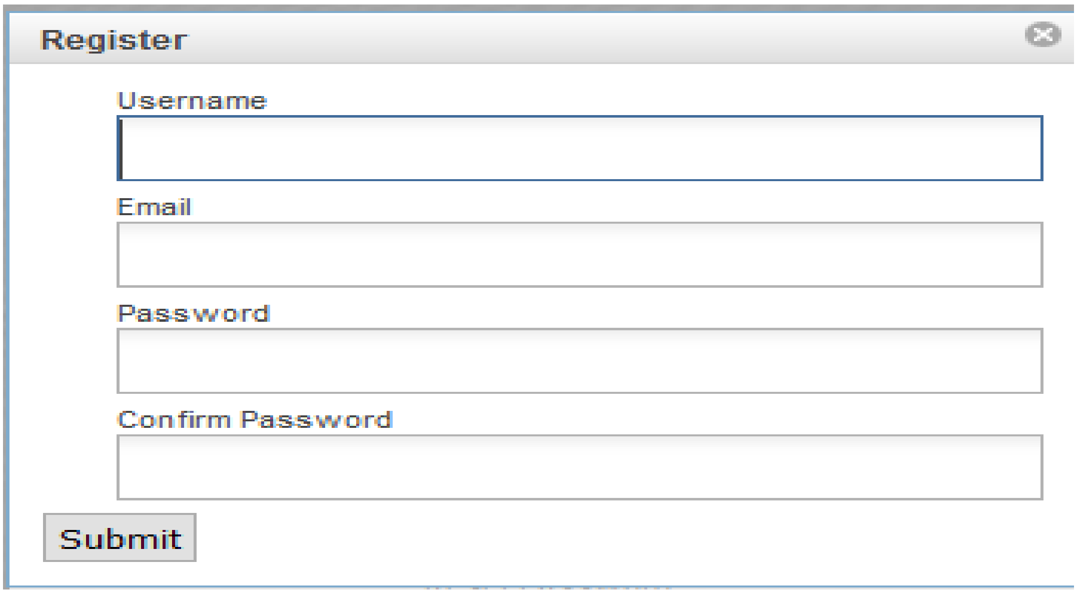
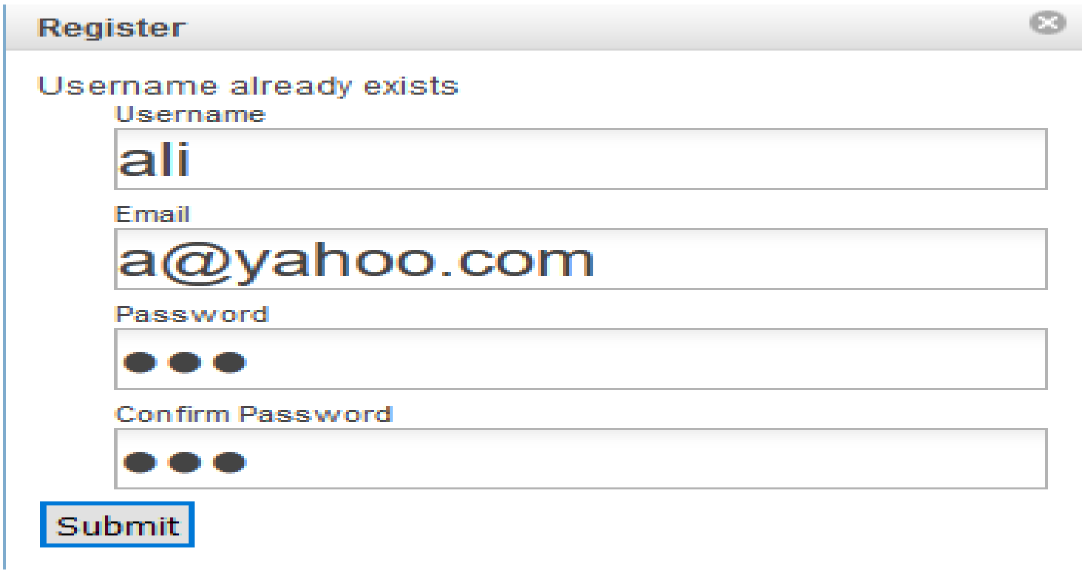


Fig (4.9) The registration page of Lucid



Username already exists

Username: ali

Email: a@yahoo.com

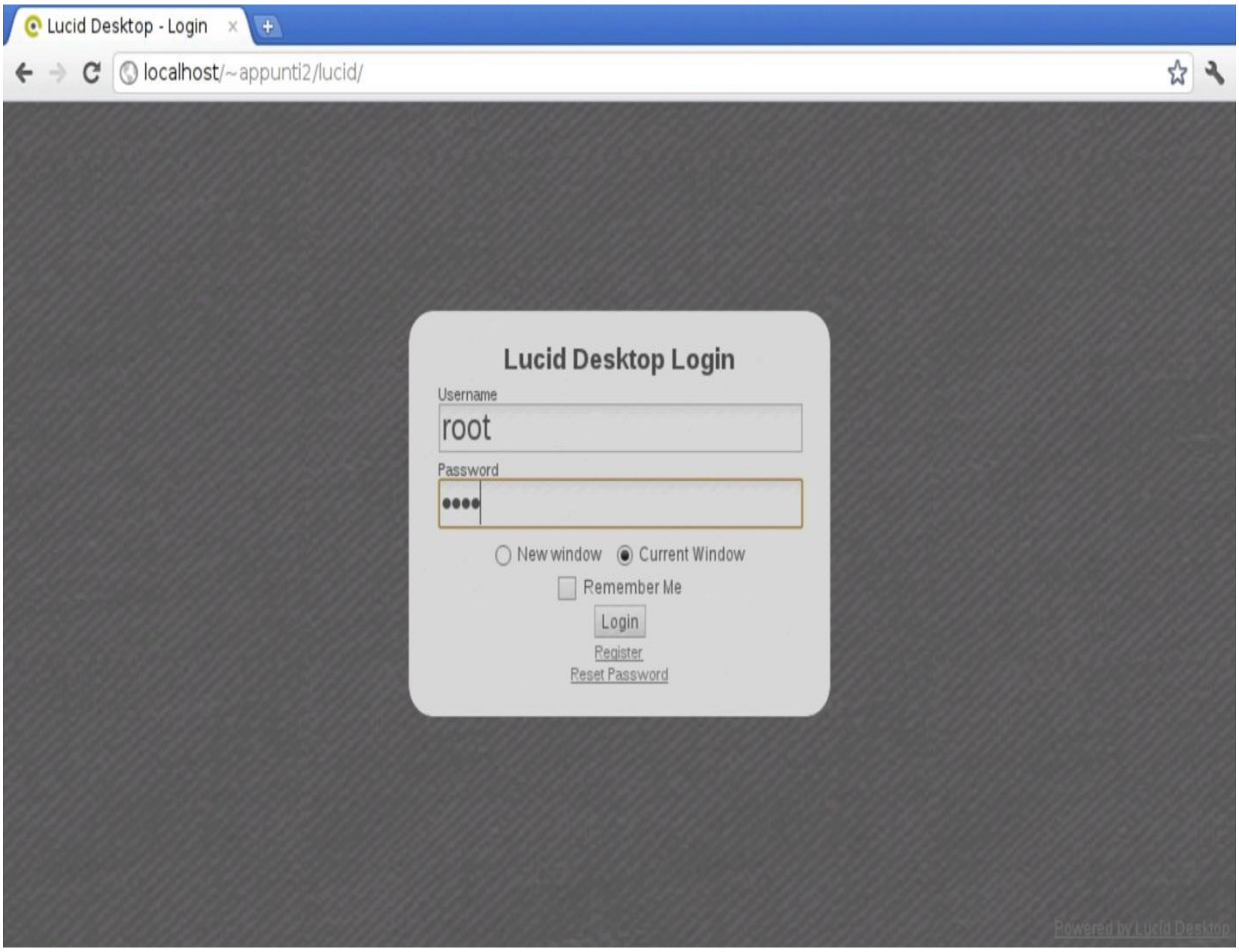
Password: ●●●

Confirm Password: ●●●

Fig (4.10) A warning message of duplicated information

A. Users Login

Logging-in is a major step that comes after registration. The registered user can access to lucid system by entering correct (username and password) Figure (4.11) shows the user login to lucid system, otherwise the system rejected



Lucid Desktop - Login

localhost/~appunti2/lucid/

Lucid Desktop Login

Username

root

Password

●●●●

○ New window ● Current Window

□ Remember Me

Login

Register

Reset Password

Powered by Lucid Desktop

Fig (4.11) the login page of lucid desktop

While a user enters own username and password which is exactly like EyeOS procedure steps, The PHP script will compare the entered username & password with those that registered in database. If they are correct, the script reads the user details such (user name, password) and then log in. Finally, the WebOS lucid index page is ready to use by the user and can see applications (i.e. games, graphics, internet, multimedia, calculator and etc.) Figure (4.12) shows the next web-page- screen that appeared the user is logging correctly.

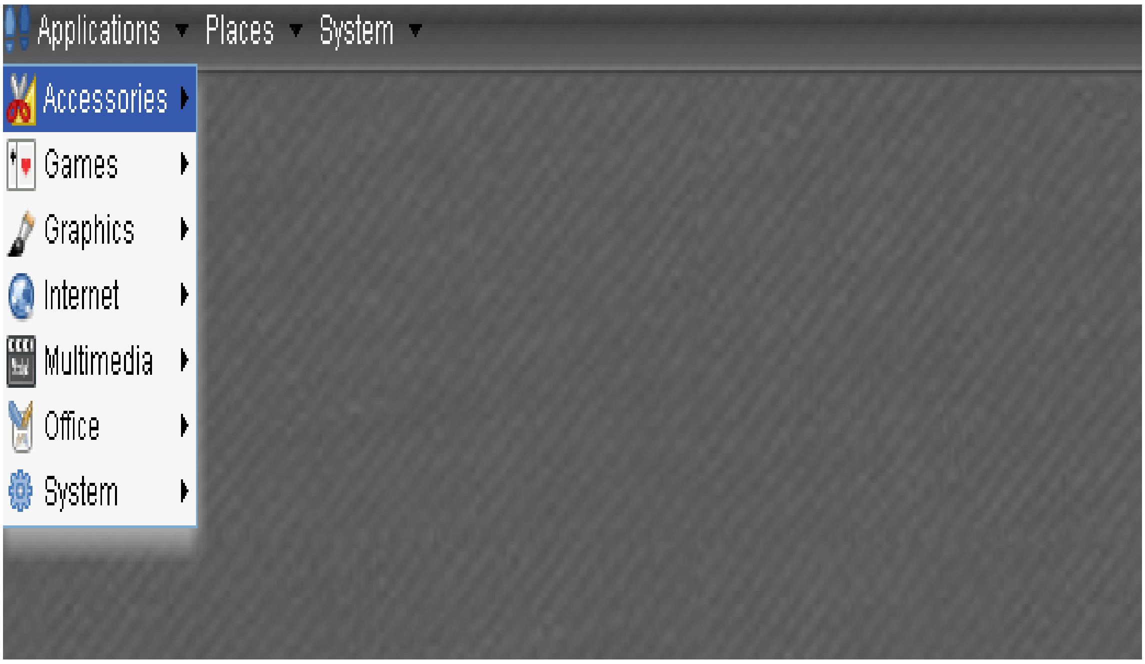
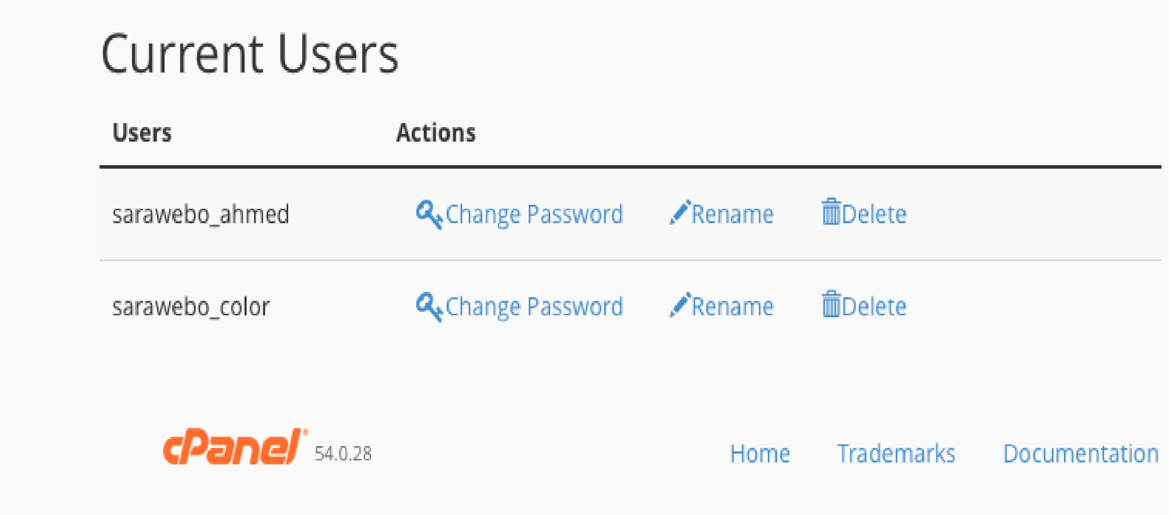


Fig (4.12) the next page after login of lucid desktop

4.4 System Administrator

For the enhanced security reason, the manually username and password is acceptable for administrating our WebOS. Thus, it is impossible to manage any WebOS without the administrator. This is a new contribution in our proposed system to have administrator. After asking the person email address who wants to become administer, the host manager offered admin (user name and password). The host manager offers the username and password for the reason to login to the system by administrator and responsible for maintaining, monitoring and managing the access of different users of system and content management. The administrator should identify himself/herself to the system by using username and password. Also, the admin is responsible for modifying and changing any information (i.e. select to delete some old job records in the system or expired jobs from the table). By the active administrator will be help to control any system (i.e. load balancing).

The admin is a backbone of every system (i.e.WebOS) and has an excellent role in security of site. Also, is control all the integrated system with full privileges. Figure (4.13) shows web page making admin to deleting the old job records.



Current Users

Users	Actions		
sarawebo_ahmed	Change Password	Rename	Delete
sarawebo_color	Change Password	Rename	Delete

cPanel 54.0.28

Home Trademarks Documentation

Fig (4.13) The webpage of current users that can be deleted from the admin

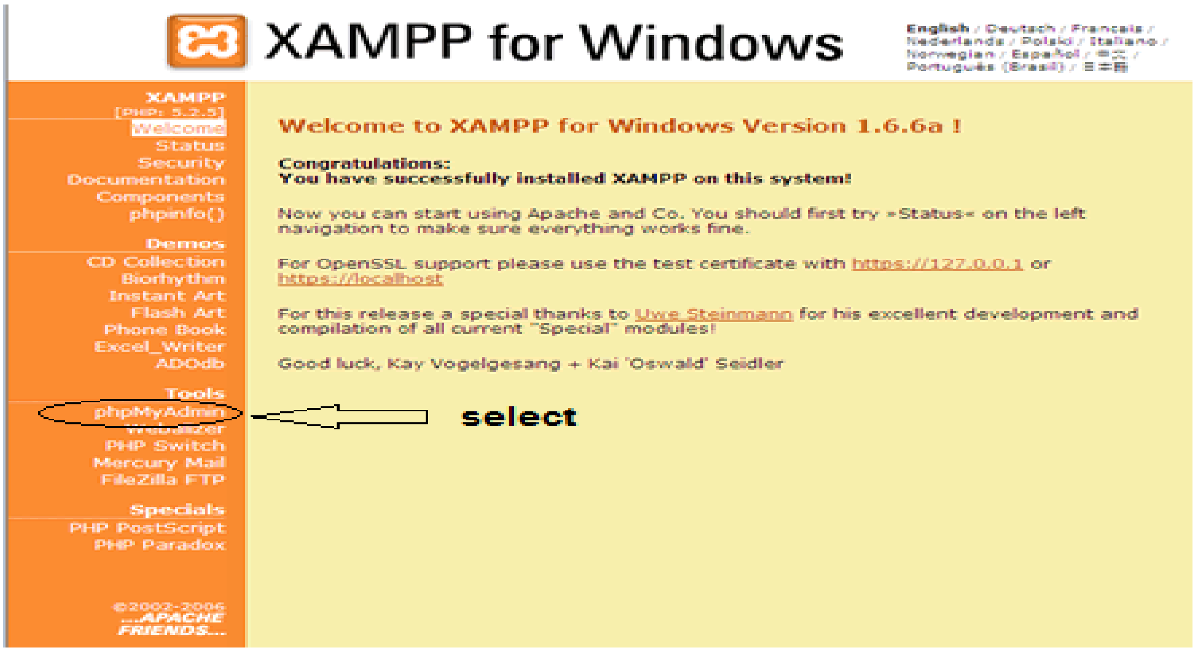
4.5: Structure of the Established Database

One of the main modules of the established WebOS is the database (DB). The established database interacts with the web pages dedicated to the individual systems (i.e. administrator and users). It include sets of data (i.e. administrator information and user's information) .In our system, the admin creates database by using the following:-

1- <u>XAMPP</u> :

To design the private WebOS, the first step of the designer must download the XAMPP on the server side and called locality database installed, hence, to download XAMPP over operating system in the server side. After opening the layout of XAMPP and selecting the "phpmyAdmin hyperlinked icon".

Figure (4.14) shows the main webpage of XAMPP that appeared phpmyAdmin .From the **"phpmyAdmin"** select a "**New**" that exists in hand side of webpage and click the **"create database"** button to name the of DB. Figure (4.15) shows the webpage of phpmyAdmin to create DB. The database created in the PrWebOS is locally and store it in backend database server. PrWebOS designed by the WebOS designer with local MySQL. For PrWebOS must be installed a copy of XAMPP, otherwise it is impossible to start our work. In this method, PrWebOS depends on its own XAMPP that is downloaded manually and it is locally.



Welcome to XAMPP for Windows Version 1.6.6a !

Congratulations:
You have successfully installed XAMPP on this system!

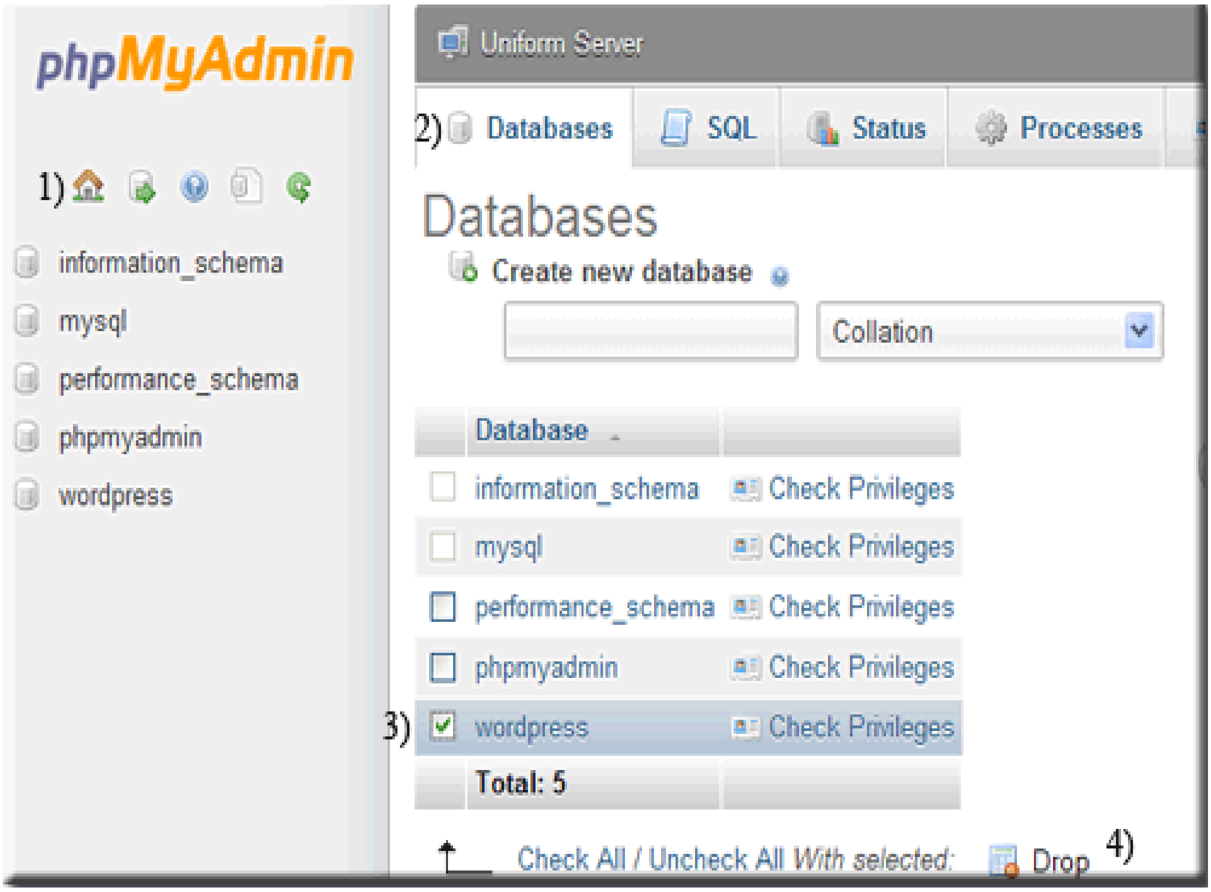
Now you can start using Apache and Co. You should first try »Status« on the left navigation to make sure everything works fine.

For OpenSSL support please use the test certificate with https://127.0.0.1 or https://localhost

For this release a special thanks to Uwe Steinmann for his excellent development and compilation of all current "Special" modules!

Good luck, Kay Vogelgesang + Kai 'Oswald' Seidler

Fig (4.14) the main webpage of XAMPP



	Database	
□	information_schema	Check Privileges
□	mysql	Check Privileges
□	performance_schema	Check Privileges
□	phpmyadmin	Check Privileges
☑	wordpress	Check Privileges
	Total: 5	

Fig (4.15) the page of phpmyAdmin to create DB

2- <u>cpanal</u>

In order to design PuWebOS the designer depends on the host that hires and becomes a global-database utility. Firstly, the designer must be an access to **cpanal** by typing the domain with cpanel such http://sarawebos.com/cpanel in address bar of the web browser. From cpanal, select the database section and then choose the database in the database section and then double click on choose "MySQL database wizard" to create database. Figure (4.16) shows the interface of cpanal.

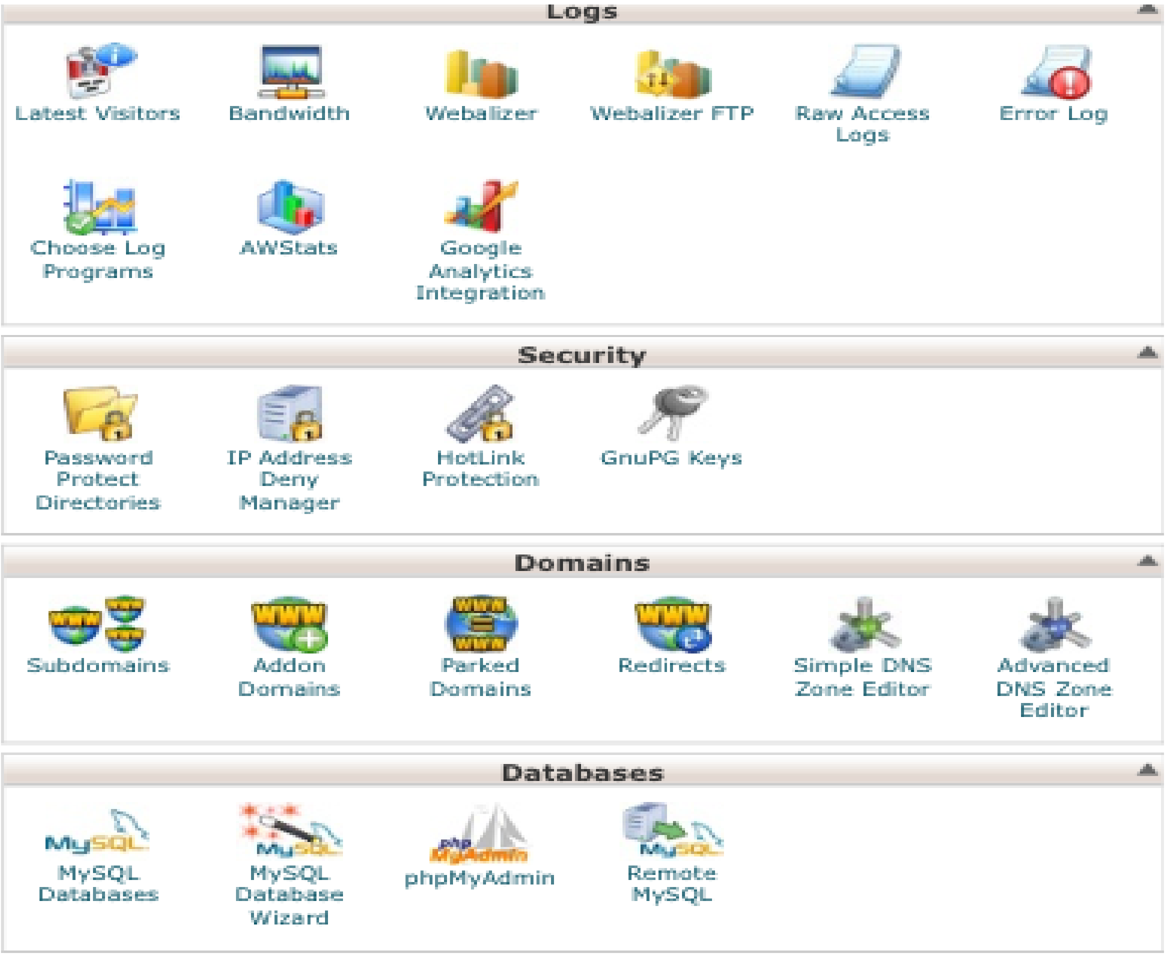


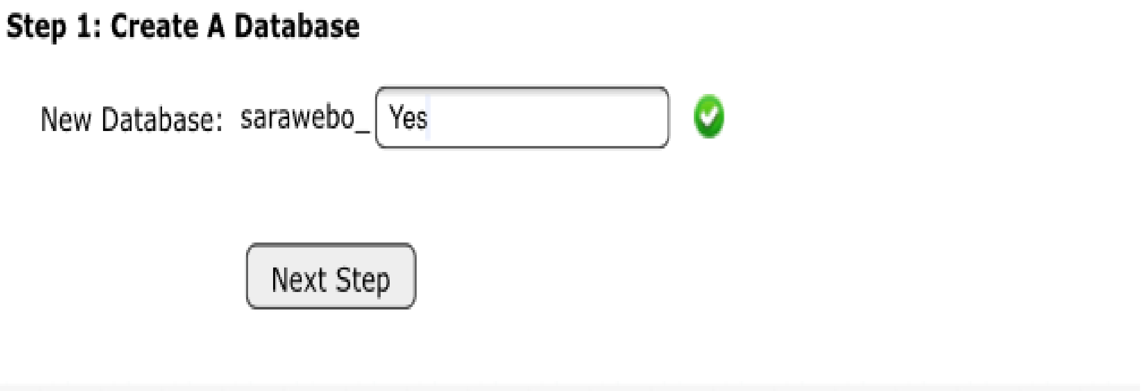
Fig (4.16) the main interface of cpanal

There are lots of application tools that related to design WebOS. The database is absolutely global because there is no need for the designer to download application tools except web browser in the computer designer and internet facility to connect the client and the server. The following are the steps to create backend data base by global database:

A- step-1 to create DB Create,

DB users by fill-up the **"username" "password"** and "**confirm Password"** button, then click create user to do.

Figure (4.17) shows the webpage to create DB users

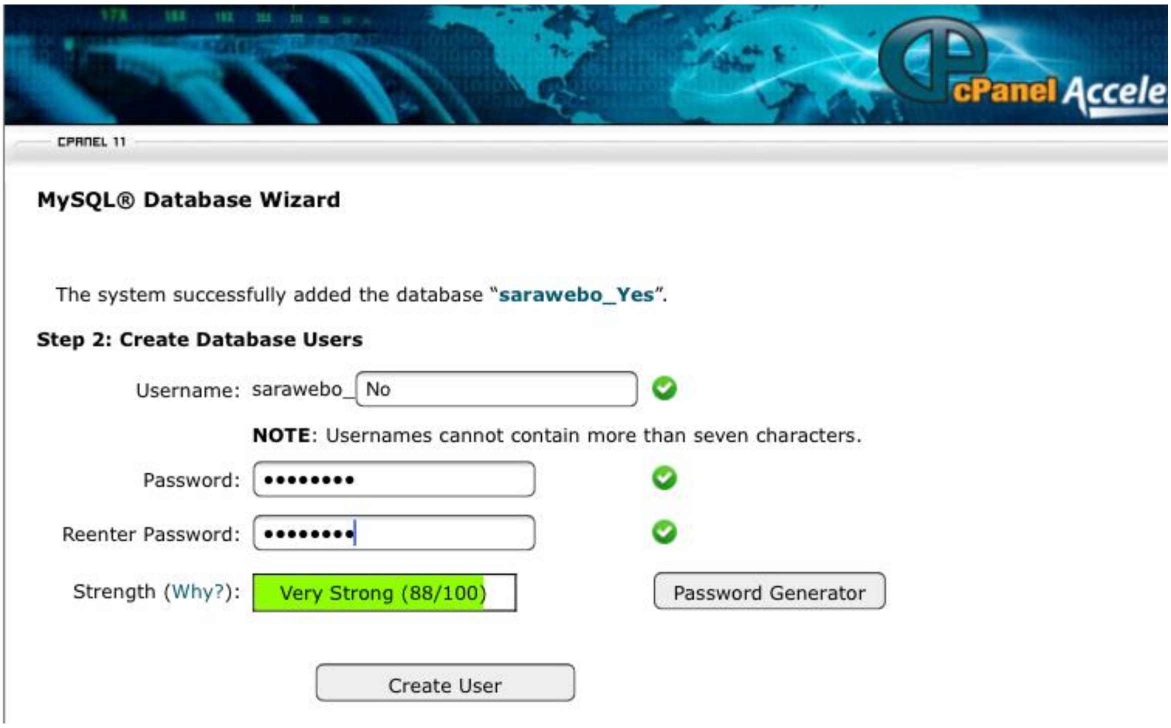


Step 1: Create A Database

New Database: sarawebo_ Yes ✓

Fig (4.17) The page of create DB

B- The step- 2 has to create DB users after the system successfully added the DB that is formerly created in step-1. Figure (4.18) shows the webpage to create DB user.



MySQL® Database Wizard

The system successfully added the database "sarawebo_Yes".

Step 2: Create Database Users

Username: sarawebo_ No ✓

NOTE: Usernames cannot contain more than seven characters.

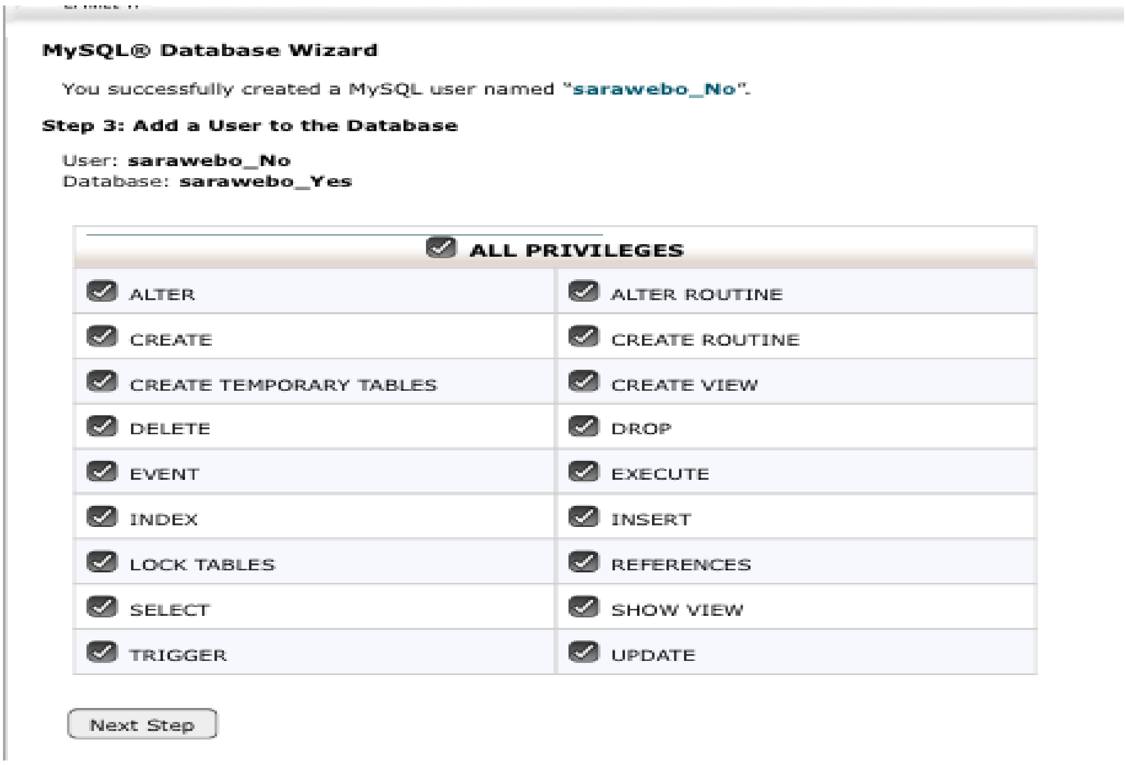
Password: •••••••• ✓

Reenter Password: •••••••• ✓

Strength (Why?): Very Strong (88/100)

Fig (4.18) the page to create database users

C- In step-3 to add user to DB, click on **"all privileges"** by √
Figure (4.19) shows the webpage after adding user to DB



MySQL® Database Wizard

You successfully created a MySQL user named "**sarawebo_No**".

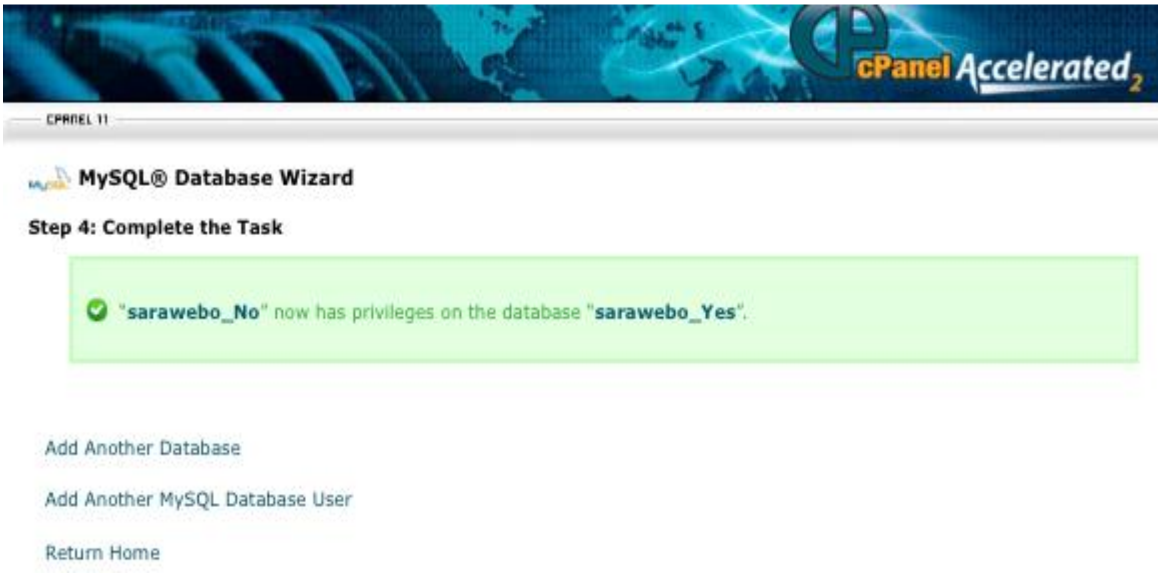
Step 3: Add a User to the Database

User: **sarawebo_No**
Database: **sarawebo_Yes**

☑ **ALL PRIVILEGES**	
☑ ALTER	☑ ALTER ROUTINE
☑ CREATE	☑ CREATE ROUTINE
☑ CREATE TEMPORARY TABLES	☑ CREATE VIEW
☑ DELETE	☑ DROP
☑ EVENT	☑ EXECUTE
☑ INDEX	☑ INSERT
☑ LOCK TABLES	☑ REFERENCES
☑ SELECT	☑ SHOW VIEW
☑ TRIGGER	☑ UPDATE

Fig (4.19) the page that add user to DB

D- The last step-4 is complete task and already the user has in DB. Figure (4.20) shows the complete task for crating DB successfully.



MySQL® Database Wizard

Step 4: Complete the Task

✓ "**sarawebo_No**" now has privileges on the database "**sarawebo_Yes**".

Fig (4.20) the page of complete creating DB successfully.

4.6 Loading-Time Analysis Implementation Criteria

Traditional loading time (TLT) is a primitive method which was before the era of ICT, but WebOS is subsequently the ICT era or new technology revolution. The technologies modify the TLT to virtually loading time (VLT) via development of communication tools. It is impossible to analyze the loading time by TLT due to changeability of human mode or activity, but VLT; for example the WebOS loading time is very easy to analyze, however the technologies and loading time together have a direct relation between them because the better capacity and speed tools reduce loading time. Therefore, the results of experiment are important to compare the loading time of different technologies with the example of PrWebOS and PuWebOS over different web operating systems (such as windows or Linux). VLT analysis implemented for two methods; which are PrWebOS and PuWebOS. A copy of the proposed lucid and EyeOS systems run on the server side of the computer - WIRELESS-LAN, for the reason of testing and find out optimal loading time. In the proposed, the LP link router is a communication tool that used for designing a wireless LAN called PrWebOS. The Same server runs lucid and EyeOS. Lucid-PrWebOS and EyeOS-PrWebOS are tested by PHP codes in order to find out the loading time. The developed criteria in this thesis are established WebOS system by using different methods namely; PrWebOS –Windows-7 operating system and PuWebOS-Linux operating system. The implementation results will be obtained using two types of OS (Windows and Linux). Finally, the results of this system will be evaluated by comparing them with a hand writing scripts in PHP (loading time). The evaluation placed in the computer-wireless using computers as client and a powerful computer as servers. The clients have different computer description (i.e. IPad, laptop, mobile and etc.) and connected via WLAN.

One time is PrWebOS and the other time is PuWebOS. The test is to find out which loading time is better when dealing with PrWebOS (takes less loading time information from the database or into database). The investigation tests of developed criteria record shown in detail in the following coming section.

4.6.1 Loading -Time Analysis styles

By means of loading time PHP codes for the way to determine how long it takes a WebOS load from the server. The writing script code by programmer and place the code which shown in Figure (3.21) at the very top of your WebOS types in PHP page (if one would like to analyze the loading time needed for particular part of the code put this right before that PHP code part) at the bottom of the page using microtime( ). PHP function you will know exactly how much time is needed for your PHP code to be loaded.

```
$mtime = microtime();
$mtime = explode(" ",$mtime); $mtime = $mtime[1] + $mtime[0];
$starttime = $mtime;

;?>

Put the codes of WebOS types and HTML in here

$mtime = microtime();
$mtime = explode(" ",$mtime);
$mtime = $mtime[1] + $mtime[0]; $endtime = $mtime;
$totaltime = ($endtime - $starttime);
echo "This page was created in ".$totaltime."

seconds"; ;?>
```

Fig (4.21) PHP script code for loading – time

4.6.2 Windows Operating System with WebOS types

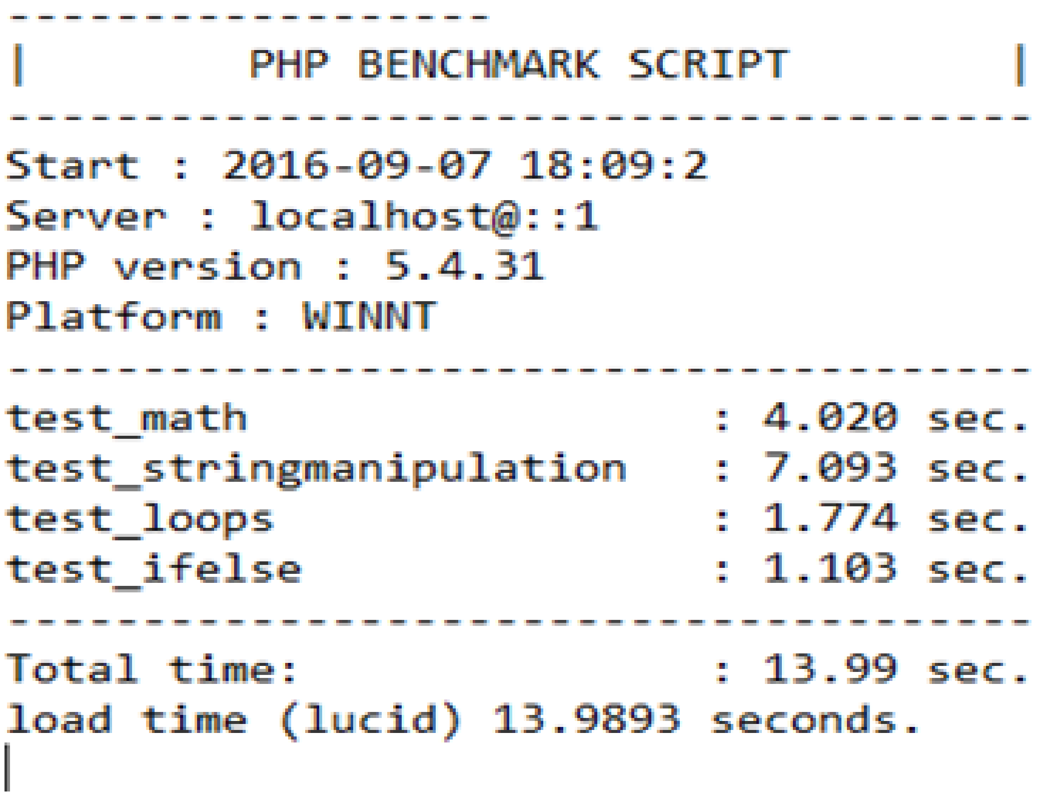
This section deals with the implementation results when using Window-7 OS with both lucid and EyeOS. The types of WebOS in (section 3.5.1) and (3.5.2) must run to test and find out individually.

4.6.2.1 Windows -7 with Lucid Desktop

In order to obtain an excellent and accurate results; there will be different groups of client samples that depend on the WebOS types.

The sample-1 is a client that is directly connected to the server in a Wireless PrWebOS for lucid .in our the proposed system. The application run on the server and with browsers the client can an access and open applications. The lucid WebOS running in three different positions:

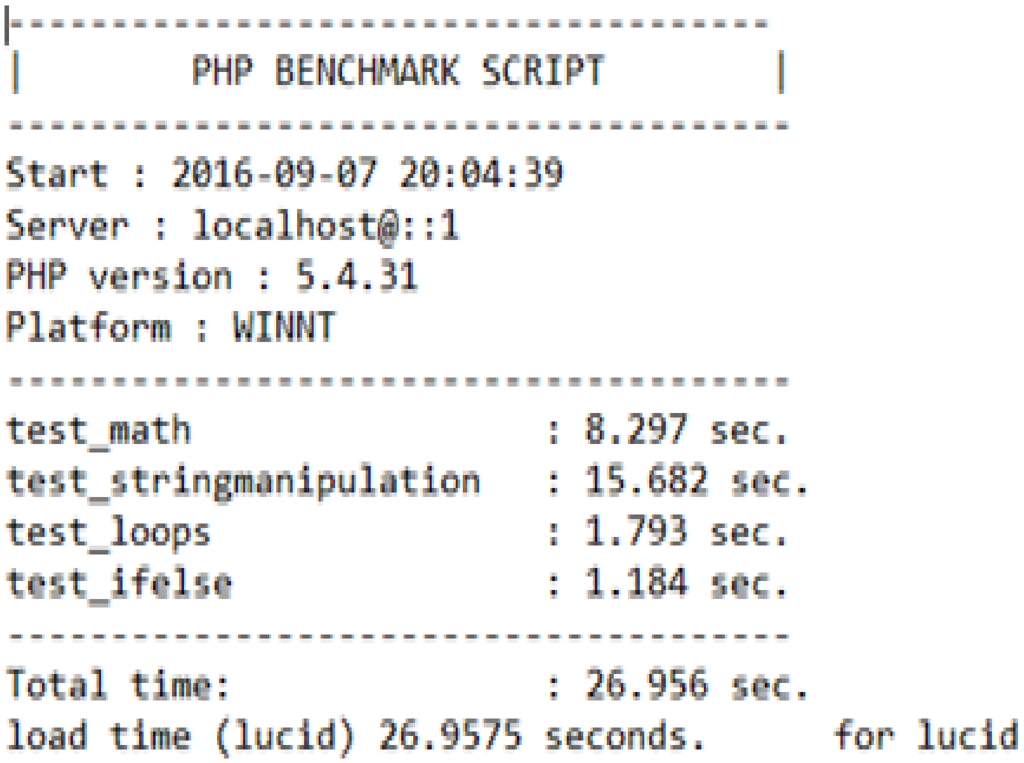
A- **Lucid-TOS:** Any single computer which is not in a net-centric (neither private nor public) and once becoming a client another a server for simulation. The installed of XAMPP software over TOS with running of our lucid proposed system in order to PHP Benchmarking Performance Script. Figures (3.22) PHP Benchmark Script for TOS results non net-centric. The codes for installing lucid WebOS are illustrated in **Appendix A**. The total time for benchmarking by PHP scripting over TOS for Lucid-PrWebOS. The result is for a single computer and not in the net and the testing is by seconds.



```
------------------
|         PHP BENCHMARK SCRIPT         |
--------------------------------------
Start : 2016-09-07 18:09:2
Server : localhost@::1
PHP version : 5.4.31
Platform : WINNT
--------------------------------------
test_math                  : 4.020 sec.
test_stringmanipulation    : 7.093 sec.
test_loops                 : 1.774 sec.
test_ifelse                : 1.103 sec.
--------------------------------------
Total time:                : 13.99 sec.
load time (lucid) 13.9893 seconds.
|
```

Fig (4.22) PHP benchmark Script for lucid over TOS results and non-net centric

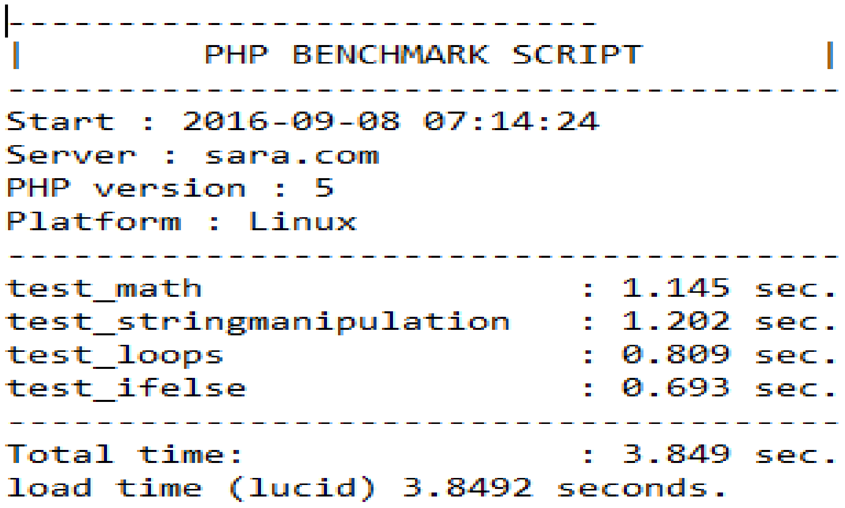
B- Lucid-PrWebOS: Any computer which is in a net-centric (either private or public) and once becomes a client must be connected to the server wireless private network as a client/server based. The installation of XAMPP software over the server is to make it dynamic. The proposed application must be installed on a server side with windows OS. The running of the lucid proposed system in order to PHP Benchmark Performance Script. Figures (3.23) shows the PHP benchmark script for private net-centric OS results. The total time of PHP benchmark script over net-centric windows OS for Lucid- PrWebOS .The result is for a single computer in the C/S net and the testing is by seconds.



```
|-----------------------------------
|        PHP BENCHMARK SCRIPT       |
------------------------------------
Start : 2016-09-07 20:04:39
Server : localhost@::1
PHP version : 5.4.31
Platform : WINNT
------------------------------------
test_math                : 8.297 sec.
test_stringmanipulation  : 15.682 sec.
test_loops               : 1.793 sec.
test_ifelse              : 1.184 sec.
------------------------------------
Total time:              : 26.956 sec.
load time (lucid) 26.9575 seconds.     for lucid
```

Fig (4.23) PHP benchmark script for lucid PrWebOS over private net- centric windows OS server.

C- Lucid-PuWebOS: Any computer which is in a net-centric (either private or public) and once becoming a client must be connected to server side as a client/server based by internet facility with the URL: http://sarawebos.com. The XAMPP software is installed over the server so as to make it dynamic. The proposed application must be uploaded by FTP FileZilla to server side with Linux OS. The uploaded of our lucid proposed system in order to PHP benchmarking performance script. Figure (3.24) shows PHP benchmark script for public net-centric OS results. The total time for benchmarking by PHP scripting over net-centric Linux OS for Lucid-PuWebOS. The result is for a single computer in the C/S net and testing with seconds.



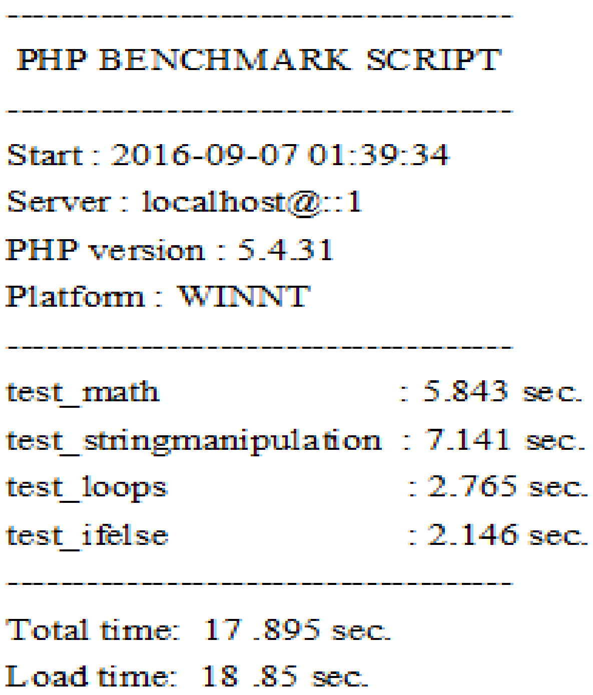
```
|--------------------------
|          PHP BENCHMARK SCRIPT          |
--------------------------------------
Start : 2016-09-08 07:14:24
Server : sara.com
PHP version : 5
Platform : Linux
--------------------------------------
test_math                    : 1.145 sec.
test_stringmanipulation      : 1.202 sec.
test_loops                   : 0.809 sec.
test_ifelse                  : 0.693 sec.
--------------------------------------
Total time:                  : 3.849 sec.
load time (lucid) 3.8492 seconds.
```

Fig (4.24) PHP benchmark script for lucid PuWebOS over public net-centric Linux OS server.

4.6.2.2 Windows -7 with EyeOS

The same test in lucid with the same phases will be repeated for EyeOS as mentioned, there will different groups of client that are depended onnWebOS types. The sample -1 is namely in lucid. The sample -2 is containing clients that are connected to the server in wireless PrWebOS for EyeOS. The proposed system application runs on server. The client can be access and open applications. There are three different types of testing which evaluated of EyeOS.

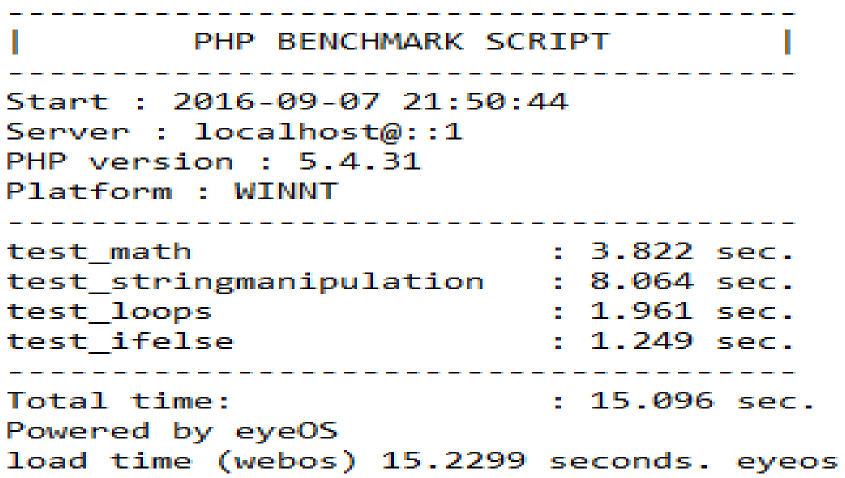
A- **EyeOS-TOS:** Any single computer which is not in net-centric (neither Private nor public) and once becomes client another time will be a server for simulation XAMPP software installed over TOS for running EyeOS proposed system in order to the PHP benchmark performance script. Figure (3.25) shows the PHP benchmark script to EyeOS over non net centric. EyeOS are illustrated in **Appendix B**. The total time of PHP benchmarking script over TOS for the EyeOS -PrWebOS. The result is for a single computer and the testing is by seconds.



```
------------------------------------
 PHP BENCHMARK SCRIPT
------------------------------------
Start : 2016-09-07 01:39:34
Server : localhost@::1
PHP version : 5.4.31
Platform : WINNT
------------------------------------
test_math                  : 5.843 sec.
test_stringmanipulation  : 7.141 sec.
test_loops                  : 2.765 sec.
test_ifelse                 : 2.146 sec.
------------------------------------
Total time:  17 .895 sec.
Load time:  18 .85 sec.
```

Fig (4.25) PHP benchmark script for EyeOS over non- net-centric windows OS server.

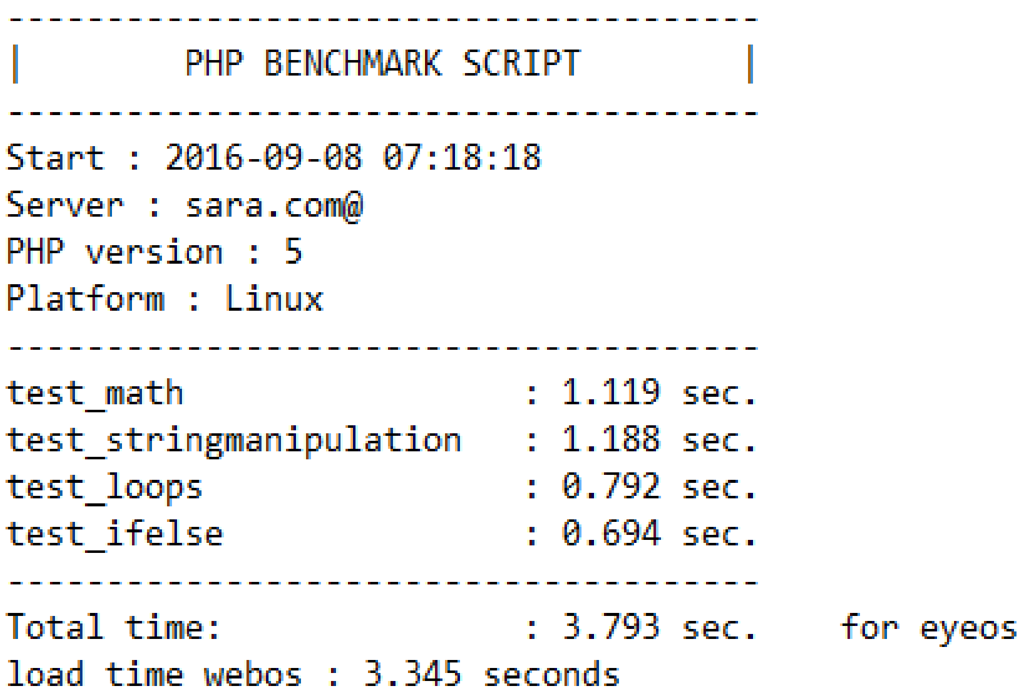
B- EyeOS - PrWebOS: Any computers which are in a net-centric (either private or public) and once becomes a client must be connected to the server wireless private network as a Client/Server based. The XAMPP software is installation over the server to make it dynamic. The proposed application must be installed on server side with windows OS. The running of the EyeOS proposed system in order to PHP benchmark performance Script. Figure (4.26) shows the PHP benchmark script for private net-centric OS results. The total time of PHP benchmark script over net- centric windows OS for EyeOS- PrWebOS. The result is for a single computer in the C/S net and the testing in by seconds.



```
-------------------------------------
|           PHP BENCHMARK SCRIPT     |
-------------------------------------
Start : 2016-09-07 21:50:44
Server : localhost@::1
PHP version : 5.4.31
Platform : WINNT
-------------------------------------
test_math                  : 3.822 sec.
test_stringmanipulation    : 8.064 sec.
test_loops                 : 1.961 sec.
test_ifelse                : 1.249 sec.
-------------------------------------
Total time:                : 15.096 sec.
Powered by eyeOS
load time (webos) 15.2299 seconds. eyeos
```

Fig (4.26) PHP Benchmark Script for EyeOS PrWebOS over private net- centric windows on server

C- **EyeOS- PuWebOS:** Any computers which are in a net-centric (either private or public) and once becomes a client must be connected to server side as a client/server based by internet facility with the URL: http://sarawebos.com. The XAMPP software is installation over the server to make it dynamic. The proposed application must be uploaded by FTP FileZilla to server side with Linux OS. The uploaded of the EyeOS proposed system in order to PHP benchmarking performance script. Figures (3.27) PHP Benchmark Script for public net-centric OS results. The total time of by PHP benchmark script over net-centric Linux OS for EyeOS-PuWebOS. The result is for a single computer in the C/S net and the testing in seconds.



```
--------------------------------------
|         PHP BENCHMARK SCRIPT       |
--------------------------------------
Start : 2016-09-08 07:18:18
Server : sara.com@
PHP version : 5
Platform : Linux
--------------------------------------
test_math                  : 1.119 sec.
test_stringmanipulation    : 1.188 sec.
test_loops                 : 0.792 sec.
test_ifelse                : 0.694 sec.
--------------------------------------
Total time:                : 3.793 sec.    for eyeos
load time webos : 3.345 seconds
```

Fig (4.27) PHP benchmark script for EyeOS- PuWebOS over net-centric Linux OS server

Table (4.1) shows the total time and load time of (Lucid and EyeOS) seconds

WebOS type	Part	Total time	Load time
Lucid	**TOS** **PrWebOS** **PuWebOS**	**13.99 sec.** **26.956 sec.** **3.84 sec.**	**13.98 sec.** **26.957 sec.** **3.84sec.**
EyeOS	**TOS** **PrWebOS** **PuWebOS**	**17.895 sec.** **15.09 sec.** **3.79 sec.**	**18.8 sec.** **15.2299 sec.** **3.34 sec.**

4.7: A New Paradigm: Load-Time and Dataflow of WebOS

Load -Time measurement by PHP script benchmarking codes for TOS non net-centric is inappropriate and inconvenient due to it is neither local nor global but it is a single user computer. However, a measurement in terms of load-time changes from a person to another request, thus the load-time is undependable. While the load time is changeable in TOS, the test regular load time criteria are impossible to occur accurately, and also the matchmaking by the users will be inaccurate.

In the proposed system WebOS over different operating systems, a paradigm of load- time has an important role for (offline and Online) testing load criteria. It means that the load-time and total time are two factors that directly related to each other with only a client computer. For example, when the load- time decreases, the system quality of performance (QoP) will be increased. The QoP for two different methods of net-centric namely PrWebOS and PuWebOS for EyeOS are much less than the lucid. Whilst the load time is reduced the value of QoP will increase. Nevertheless, EyeOS is much better result if compared to lucid. The results are shown in table (4.1). However, the Non net-centric means the TOS for EyeOS and lucid experiment result show the lucid load time is less than EyeOS, from that the QoP of lucid is better for non-net-centric. Hence the net-centric WebOS (i.e. EyeOS) has a better result than lucid. As mentioned the results are shown in table (4.1), But of both case the PuWebOS for (EyeOS and lucid) is better than (PrWebOS and TOS).

Chapter Five
Conclusions and Suggestions

5.1 Conclusion

In this thesis, methods for WebOS are designed, implemented and tested. Some of the recommendation methods are based on the matching networking type (i.e. net-centric and non-net-centric) with types of WebOS WebOS (i.e. PrWebOS, PuWebOS and TOS).

During the design and implementation phases of the proposed WebOS, the following remarks have been addressed:

1- The proposed WebOS can improve the communication benefits between web technology and operating system. WebOS is a new generation between Traditional operating system and web technologies. WebOS is a faster and more reasonable way with an execution user request if comparison with of using depend TOS.

2- The Private and public WebOS are based networks can access resources at less fine loading than TOS based network does.

3- There are many intelligent WebOS methods (EyeOS, Lucid) and they will give different loading time for the same case.

4- The WebOS is more advanced than OS and it is more accurate and reliable for using software applications. Installed copy of the proposed system in a single personal computer (PCs); couldn’t be considered as a type of WebOS.

5- The QoP is increased when load time is decreased with web technology and distributed database management systems (including: MYSQL, PHP, HTML) are used.

6- One of the advantages of the proposed system is it easy and updateable at any time there is need to change the applied rules. Also, the benefits of using WebOS are cost effective. In real life and make life very easy.

5.2 Suggestions for Future Works

During the implementation and testing phases, many ideas have been stimulated as proposals for further works to improve the Quality of performance of the developed WebOS types. Some of these proposals are listed below:-

1- Using WebOS instead of TOS and adding extra system services. For example, placing voice or video chatting, picture exchange and SMS.

2- Instead of using only the individual OS in dependently that gets more inaccurate and unreasonable results.

3- Adding extra services with Internet users get the benefits of the proposed Internet-based WebOS application.

4- Improving the used WebOS to speed up the system performance and to be capable to handle many of the individual attributes as WebOS.

Appendix A

Lucid desktop

```
php؟>
؛()time = microtime$
؛(time = explode(' ', $time$
؛[time = $time[1] + $time[0$
؛start = $time$
<؟
<html>
<head>
<title> WebOS Lucid</title>
script type="text/javascript" src="../desktop/dojotoolkit/dojo/dojo.js" >
<djConfig="parseOnLoad: true"></script
<script type="text/javascript" src="install.js"></script>
</ "link href="style.css" rel="stylesheet" media="screen>
link href="../desktop/dojotoolkit/dijit/themes/dijit.css" rel="stylesheet" >
</ "media="screen
link href="../desktop/dojotoolkit/dijit/themes/tundra/tundra.css" >
</ "rel="stylesheet" media="screen
link href="../desktop/dojotoolkit/dijit/themes/tundra/tundra_rtl.css" >
</ "rel="stylesheet" media="screen
<head/>
<"body class="tundra>
<"form dojoType="dijit.form.Form" id="form>
<"div id="pageBar>
<div id="pageBarOverlay"></div>
div id="wizardPages" dojoType="dijit.layout.StackController" >
<containerId="wizard"></div
<div/>
<noscript>
<"style type="text/css>
} wizard, .buttons#
؛display: none
{
```

```
<style/>
<"؛div class="wizard" style="text-align: center>
div>You must have javascript enabled to use the >
<WebOS Lucid.</div
div>Please enable javascript and refresh the page to >
<continue.</div
<div/>
<noscript/>
div dojoType="dijit.layout.StackContainer" id="wizard" >
<"widgetid="wizard
div dojoType="dijit.layout.ContentPane" title="Start" >
<"id="start
div class="title">Welcome to the WebOS >
<Lucid!</div
<p>
This will install and configure WebOS Lucid
<Desktop for your server.<br
or      You can use this installer to install a new
.reset your current system,
<p/>
<p>
The installer requires that you have some sort of
.DBMS backend for the desktop to use mySQL
<p/>
<div/>
div dojoType="dijit.layout.ContentPane" title="Installation >
<"Type" id="installtype-choose
div class="title">What type of installation do you >
want?</div
<
input dojoType="dijit.form.RadioButton" >
</ "id="installtype-new" type="radio" name="type" value="clean
```

```
<label for="installtype-new"><b>I want to do a clean installation</b></label>
p>Do a new installation, and create a new admin >
<user</p
div style="display: none;"> <!-- remove for now, no CMS >
<-- support
input dojoType="dijit.form.RadioButton" >
</ "id="installtype-cms" type="radio" name="type" value="CMS" disabled="true
label for="installtype-cms"> <b>I want to do a clean >
<installation, but intergrate it with a CMS</b></label
p>Do a new installation, then configure it to intergrate >
<with a user table from a CMS</p
<div/>
We should support drupal, joomla/mambo,  --!>
<-- phpnuke, and some others
input dojoType="dijit.form.RadioButton" >
</ "id="installtype-reset" type="radio" name="type" value="reset
label for="installtype-reset"><b>I want to reset my >
<existing installation</b></label
p>Keeps your settings and users, but re-installs apps >
<and other things</p
<div/>
div dojoType="dijit.layout.ContentPane" title="Permissions" >
<"id="permissions
<div class="title">Permissions Check</div>
Before you start the installation, you need to change
</ some file permissions:<br
<div id="perms">please wait...</div>
<div/>
div dojoType="dijit.layout.ContentPane" title="Database" >
<"؛([onclick="install.fixurlStr(arguments[0]);" onkeyup="install.fixurlStr(arguments[0
<div class="title">Database Settings</div>
</ br>
:Database Backend
<div>
```

```
input dojoType="dijit.form.RadioButton" >
</ "type="radio" name="db_type" id="db_type_mysql" value="mysql
<label for="db_type_mysql"> mySQL</label>
<div/>
<div>
input dojoType="dijit.form.RadioButton" >
</ "type="radio" name="db_type" id="db_type_mysqli" value="mysqli
<label for="db_type_mysqli"> mySQLi</label>
<div/>
<div>
input dojoType="dijit.form.RadioButton" >
</ "type="radio" name="db_type" value="sqlite" id="db_type_sqlite
<label for="db_type_sqlite"> SQLite</label>
<div/>
<div>
input dojoType="dijit.form.RadioButton" >
</ "type="radio" name="db_type" value="mssql" id="db_type_mssql
label for="db_type_mssql"> Microsoft SQL >
<Server <b>[Experimental]</b></label
<div/>
<div>
input dojoType="dijit.form.RadioButton" >
</ "type="radio" name="db_type" value="pgsql" id="db_type_pgsql
label for="db_type_pgsql"> PostgreSQL >
<<b>[Experimental]</b> (broken)</label
<div/>
<"؛div style="display: none>
input dojoType="dijit.form.RadioButton" >
</ "type="radio" name="db_type" value="ibase" id="db_type_ibase
<label for="db_type_ibase"> iBase</label>
<div/>
<div>
```

```
input dojoType="dijit.form.RadioButton" type="radio" name="db_type" value="oci8" >
</ "id="db_type_oci8
label for="db_type_oci8"> Oracle DB (versions >
<7-10) <b>[Experimental]</b></label
<div/>
<div>
b>Note:</b> Experimental DBs are not officially supported. If possible, use >
.MySQL or SQLite
<div/>
</ br>
؛DB Host:&nbsp
input dojoType="dijit.form.TextBox" name="db_host" >
<type="text" /> <i style="font-size: 10pt;">(Path to database for SQLite)</i
</ br>
؛DB Name:&nbsp
input dojoType="dijit.form.TextBox" >
<name="db_name" type="text" /> <i style="font-size: 10pt;">(Not used in SQLite)</i
</ br>
؛Table Prefix:&nbsp
input dojoType="dijit.form.TextBox" >
</ "name="db_prefix" type="text
</ br>
؛Username:&nbsp
input dojoType="dijit.form.TextBox" >
<name="db_username" type="text" /> <i style="font-size: 10pt;">(Not used in SQLite)</i
</ br>
؛Password:&nbsp
input dojoType="dijit.form.TextBox" >
<name="db_password" type="text" /> <i style="font-size: 10pt;">(Not used in SQLite)</i
</ br /><br>
DSN: <input dojoType="dijit.form.TextBox"
</ name="db_url" id="urlstr" style="width: 800px;" type="text" /><br
```

```
<i style="color: red;">Do not change this unless you know what you're doing</i>
<div/>
div dojoType="dijit.layout.ContentPane" title="Admin Setup" >
<"؛()onkeyup="install.checkAdmin
<div class="title">Adminstrative User Settings</div>
؛Admin Username:&nbsp
input dojoType="dijit.form.TextBox" >
</ "name="admin_user" type="text
<"؛()div onkeyup="install.onPasswordChange>
؛Admin Password:&nbsp
input dojoType="dijit.form.TextBox" >
<name="admin_pass" type="password" /><span id="admin_passbox"></span
<div/>
<"؛()div onkeyup="install.onPasswordChange>
؛Confirm Password:&nbsp
input dojoType="dijit.form.TextBox" >
</ "name="admin_confpass" type="password
<div/>
<div>
؛Admin E-mail:&nbsp
input dojoType="dijit.form.TextBox" >
</ "name="admin_email" type="text
<div/>
<"؛div style="margin-top: 10px>
:Allow public registration
<div>
input checked >
dojoType="dijit.form.RadioButton" type="radio" name="conf_public" id="conf_public_true"
</ "value="true
label for="conf_public_true"> >
<yes</label
<div/>
<div>
```

```
input >
dojoType="dijit.form.RadioButton" type="radio" name="conf_public"
</ "id="conf_public_false" value="false
label for="conf_public_false"> >
<no</label
<div/>
<div/>
<"؛div style="margin-top: 10px>
Throttle crosstalk requests under high server
:load
<div>
input >
dojoType="dijit.form.RadioButton" type="radio" name="conf_throttle"
</ "id="conf_throttle_true" value="true
label for="conf_throttle_true"> >
<yes</label
<div/>
<div>
input checked >
dojoType="dijit.form.RadioButton" type="radio" name="conf_throttle"
</ "id="conf_throttle_false" value="false
label for="conf_throttle_false"> >
<no</label
<div/>
<div/>
<div/>
div dojoType="dijit.layout.ContentPane" title="Pre-Install" >
<"id="config-check
<div class="title">Pre-installation</div>
Checking for any conflicts, and current server
</ environment...<br
<div id="configcheck">please wait...</div>
<div id="pinstall">please wait...</div>
<div/>
```

```
div dojoType="dijit.layout.ContentPane" id="install-page" >
<"title="Installing
<div class="title">Installing...</div>
<"div id="progressContainer>
<"div id="progressBarContainer>
div dojoType="dijit.ProgressBar" >
<id="progressBar" indeterminate="true"></div
<div/>
<"div id="taskList>
...Starting installation
<div/>
<div/>
<div/>
<"div dojoType="dijit.layout.ContentPane" title="Finish>
<div class="title">Installation Completed</div>
Please delete the /install/ directory for security purposes.
You may also want to chmod /backend/configuration.php as well as /backend/ back to a lower
permission level, depending on your server's setup. If not, you can leave it as-is, but try to
!avoid it. Enjoy
<div/>
<div/>
<"div class="buttons>
button dojoType="dijit.form.Button" id="previous" >
<onClick="dijit.byId('wizard').back()">&lt;</button
button dojoType="dijit.form.Button" id="next" >
<onClick="dijit.byId('wizard').forward()">&gt;</button
<div/>
<form/>
<body/>
<html/>
php؟>
} (function test_Math($count = 140000
؛(time_start = microtime(true$
mathFunctions = array("abs", "acos", "asin", "atan", "bindec", "floor", "exp", $
؛(""sin", "tan", "pi", "is_finite", "is_nan", "sqrt
} (foreach ($mathFunctions as $key => $function
؛([if (!function_exists($function)) unset($mathFunctions[$key
{
} (++for ($i=0; $i < $count; $i
} (foreach ($mathFunctions as $function
؛((r = call_user_func_array($function, array($i$
{
{
؛(return number_format(microtime(true) - $time_start, 3
{
} (function test_StringManipulation($count = 130000
؛(time_start = microtime(true$
stringFunctions = array("addslashes", "chunk_split", "metaphone", $
؛(""strip_tags", "md5", "sha1", "strtoupper", "strtolower", "strrev", "strlen", "soundex", "ord
} (foreach ($stringFunctions as $key => $function
؛([if (!function_exists($function)) unset($stringFunctions[$key
{
؛"string = "the quick brown fox jumps over the lazy dog$
} (++for ($i=0; $i < $count; $i
} (foreach ($stringFunctions as $function
؛((r = call_user_func_array($function, array($string$
{
{
return number_format(microtime(true) - $time_start, 3
؛(
{
} (function test_Loops($count = 19000000
؛(time_start = microtime(true$
؛(for($i = 0; $i < $count; ++$i
؛i = 0; while($i < $count) ++$i$
؛(return number_format(microtime(true) - $time_start, 3
{
```

```
(function test_IfElse($count = 9000000
}
؛(time_start = microtime(true$
} (++for ($i=0; $i < $count; $i
} (if ($i == -1
} (elseif ($i == -2 {
} (else if ($i == -3 {
{
{
؛(return number_format(microtime(true) - $time_start, 3
{
؛total = 0$
؛()functions = get_defined_functions$
؛("-"،line = str_pad("-",38$
echo "<pre>$line\n|".str_pad("PHP BENCHMARK SCRIPT",36,"
",STR_PAD_BOTH)."|\n$line\nStart : ".date("Y-m-d H:i:s")."\nServer :
{$_SERVER['SERVER_NAME']}@{$_SERVER['SERVER_ADDR']}\nPHP version :
؛"".PHP_VERSION."\nPlatform : ".PHP_OS. "\n$line\n
} (foreach ($functions['user'] as $user
} ((if (preg_match('/^test_/', $user
؛()total += $result = $user$
؛"echo str_pad($user, 25) . " : " . $result ." sec.\n
{
{
echo str_pad("-", 38, "-") . "\n" . str_pad("Total time:", 25) . " : " . $total ." sec.</pre
؛"<
<؟
php؟>
؛()time = microtime$
؛(time = explode(' ', $time$
؛[time = $time[1] + $time[0$
؛finish = $time$
```

```
؛(total_time = round(($finish - $start), 4$
؛'.echo 'load time (lucid) '.$total_time.' seconds
<؟
```

Appendix B
EyeOS

```
php؟>
؛()time = microtime$
؛(time = explode(' ', $time$
؛[time = $time[1] + $time[0$
؛start = $time$
<؟
php؟>
} (function test_Math($count = 140000
؛(time_start = microtime(true$
mathFunctions = array("abs", "acos", "asin", "atan", $
؛(""bindec", "floor", "exp", "sin", "tan", "pi", "is_finite", "is_nan", "sqrt
} (foreach ($mathFunctions as $key => $function
if (!function_exists($function))
؛([unset($mathFunctions[$key
{
} (++for ($i=0; $i < $count; $i
foreach ($mathFunctions as
} ($function
r = $
؛((call_user_func_array($function, array($i
{
{
؛(return number_format(microtime(true) - $time_start, 3
{
} (function test_StringManipulation($count = 130000
؛(time_start = microtime(true$)
stringFunctions = array("addslashes", "chunk_split", $
"metaphone", "strip_tags", "md5", "sha1", "strtoupper", "strtolower", "strrev", "strlen",
؛(""soundex", "ord
} (foreach ($stringFunctions as $key => $function
if (!function_exists($function))
؛([unset($stringFunctions[$key
{
؛"string = "the quick brown fox jumps over the lazy dog$
} (++for ($i=0; $i < $count; $i
foreach ($stringFunctions as
} ($function
r = $
؛((call_user_func_array($function, array($string
{
{
؛(return number_format(microtime(true) - $time_start, 3
{
} (function test_Loops($count = 19000000
؛(time_start = microtime(true$
؛(for($i = 0; $i < $count; ++$i
؛i = 0; while($i < $count) ++$i$
؛(return number_format(microtime(true) - $time_start, 3
{
} (function test_IfElse($count = 9000000
؛(time_start = microtime(true$
} (++for ($i=0; $i < $count; $i
} (if ($i == -1
} (elseif ($i == -2 {
} (else if ($i == -3 {
{
{
return number_format(microtime(true) - $time_start, 3
     ؛(
    {
؛total = 0$
؛()functions = get_defined_functions$
؛("-"،line = str_pad("-",38$
echo "<pre>$line\n|".str_pad("PHP BENCHMARK
SCRIPT",36," ",STR_PAD_BOTH)."|\n$line\nStart : ".date("Y-m-d H:i:s")."\nServer :
{$_SERVER['SERVER_NAME']}@{$_SERVER['SERVER_ADDR']}\nPHP version :
؛"".PHP_VERSION."\nPlatform : ".PHP_OS. "\n$line\n
} (foreach ($functions['user'] as $user
} ((if (preg_match('/^test_/', $user
؛()total += $result = $user$
؛"echo str_pad($user, 25) . " : " . $result ." sec.\n
{
{
echo str_pad("-", 38, "-") . "\n" . str_pad("Total time:",
؛"<25) . " : " . $total ." sec.</pre
<؟
php؟>
} (('if (ini_get('register_globals
} (foreach (array_keys($_REQUEST) as $key
} (if ($_REQUEST[$key] === $$key
؛(unset($$key
{
{
{
Support get_magic_quotes_gpc and magic_quotes_sybase //
if ((function_exists('get_magic_quotes_gpc') && get_magic_quotes_gpc()) ||
} (('ini_get('magic_quotes_sybase
COOKIE = array_map_recursive('stripslashes', _$
؛($_COOKIE, true
GET = array_map_recursive('stripslashes', $_GET, _$
true
```

```
؛(
POST = array_map_recursive('stripslashes', $_POST, _$
؛(true
REQUEST = array_map_recursive('stripslashes', _$
؛($_REQUEST, true
{
} (('if (ini_get('magic_quotes_runtime') && function_exists('set_magic_quotes_runtime
؛(set_magic_quotes_runtime(0@
{
} (function array_map_recursive($callback, $array, $mapkeys = false
} ((if ($callback !== null && is_callable($callback
} (foreach ($array as $key => $value
} (if ($mapkeys
؛([unset($array[$key
key = call_user_func($callback, $
؛($key
{
} ((if (is_array($value
array[$key] = $
؛(array_map_recursive($callback, $value
} else {
array[$key] = $
؛(call_user_func($callback, $value
{
{
{
؛return $array
{
؛(define('EYE_INDEX',1
؛('require_once('settings.php
؛()changeCWD
؛()loadStringLibrary
include_once(EYE_ROOT.'/'.SYSTEM_DIR.'/'.KERNEL_DIR.'/kernel'.EYE_CODE_EXTE
NSION
```

```
؛(
؛()setPhpInitDebug
؛()setPhpInitValues
؛()libraryLoading
؛()serviceLoading
؛()index = indexRequested$
}(if($index !== false
؛(loadIndex($index
}(()elseif(clientMobile{
} (()if (mobileWithWebkit
؛('loadIndex('iphone
} else {
؛('loadIndex('mobile
{
}else{
؛('loadIndex('browser
{
}(function loadIndex($index
If some index has been loaded, return false because //
indexes can't be mixed
}(('if(defined('INDEX_TYPE
؛('/../.'،'define('INDEX_BASE
؛return false
{
؛('/.'،'define('INDEX_BASE
Include the file with the __FILE__ secure//
؛'/'.((__myPath = dirname(realpath(__FILE$
؛('rPath = realpath($myPath.'/'.$index.'/index.php$
}((if(is_readable($rPath
؛(require_once($rPath
؛return true
{
؛return false
```

```
{
}()function indexRequested
if(isset($_REQUEST['index']) &&
}((['!empty($_REQUEST['index
؛(['return utf8_basename($_REQUEST['index
{
؛return false
{
*/
.(Check if the client is a cell phone without special support (like iphone*
/*
}()function clientMobile
} (if(CHECK_MOBILE == 1
mobileClients = array$
،"midp"
،"x320٢٤٠"
،"blackberry"
،"netfront"
،"nokia"
،"panasonic"
،"portalmmm"
،"sharp"
،"-sie"
،"sonyericsson"
،"symbian"
،"windows ce"
،"benq"
،"mda"
،"-mot"
،"opera mini"
،"philips"
،"pocket pc"
```

```
،"sagem"
،"samsung"
،"sda"
،"-sgh"
،"vodafone"
،"xda"
"iphone"
؛(
userAgent = $
؛(['utf8_strtolower($_SERVER['HTTP_USER_AGENT
} (foreach($mobileClients as $mobileClient
if (strstr($userAgent,
} (($mobileClient
؛return $mobileClient
{
{
؛return false
{
{
} ()function mobileWithWebkit
userAgent = $
؛(['utf8_strtolower($_SERVER['HTTP_USER_AGENT
if (strstr($userAgent, 'webkit') || strstr($userAgent,
} (("android
؛return true
} else {
؛return false
{
{
*/
Load the utf8 support loading eyeString and making a fake*
.(load because kernel needs it (utf8 support*
/*
}()function loadStringLibrary
include_once(EYE_ROOT.'/'.SYSTEM_DIR.'/'.LIB_DIR.'/eyeString/main'.EYE_CODE_EX
؛(TENSION
؛('call_user_func('lib_eyeString_start
setting library loaded//
؛(define('LIB_EYESTRING_LOADED',1
{
*/
Set the eyeOS debuggin, at the moment only changes*
.the error_reporting, but may change more things in the future*
/*
}()function setPhpInitDebug
؛(ini_set('html_errors', false
} (if(EYEOS_DEBUG_MODE == 0
؛(error_reporting(0
؛(ini_set('display_errors', false
} (elseif(EYEOS_DEBUG_MODE == 2 {
؛(error_reporting(E_ALL
؛(ini_set('display_errors', true
} (elseif(EYEOS_DEBUG_MODE == 3 {
} (('if (!defined('E_DEPRECATED
؛(define('E_DEPRECATED', 0
{
error_reporting(E_ALL ^ E_DEPRECATED ^
؛(E_NOTICE
؛(ini_set('display_errors', true
} else {
؛(error_reporting(E_ERROR
؛(ini_set('display_errors', true
{
{
*/
Load the basics libraries needed by the kernel/core*
/*
```

```
}()function libraryLoading
Loading the Error Codes//
؛('reqLib('errorCodes','loadCodes
load pear library class//
؛('reqLib('eyePear','loadPear
{
*/
Load the basic services needed by the kernel/core*
/*
}()function serviceLoading
Loading the Security Service (sec) if eyeOS Security is //
(turned on (by default is On
} (if(EYEOS_SECURITY == 1
؛('service('sec','start
{
Setting the Running Log check var to 0//
؛global $LOG_RUNNING
؛LOG_RUNNING = 0$
{
*/
Set some php init values depending of eyeOS configs*
/*
}()function setPhpInitValues
if allow_big_streams php will not have //
max_execution_time
} (if(ALLOW_BIG_STREAMS == 1
؛(set_time_limit(0@
{
set the default charset//
؛(ini_set('default_charset', DEFAULT_CHARSET
{
*/
Changes the current work directory to EYE_ROOT*
/*
}()function changeCWD
since index.php is always below eyeROOT, we can do //
this instead to be inclusable from third party code
؛'/'.(__basedir = dirname(__FILE$
change directory to EYE_ROOT//
؛(chdir($basedir.REAL_EYE_ROOT
Loaded before kernel for kernel utf8 compatibility//
{
<؟
php؟>
؛()time = microtime$
؛(time = explode(' ', $time$
؛[time = $time[1] + $time[0$
؛finish = $time$
؛(total_time = round(($finish - $start), 4$
؛'.echo 'load time (webos) '.$total_time.' seconds
 <؟
```

References

[1] Ranjan K, M, Debabrata.S, "Web OS in Cloud Computing: A Case study" *Int. J. Advanced Networking and Applications,* Volume: 0, Issue:05, Pages:2868-2875, ISSN: 0975-0290. (2016).

[2] Amin V., E. Belani, P.Eastham, C.Yoshikawa, T.Anderson, D.Culler, M.Dahlin, "webos: operating system services for wide area Application", Computer science Department, University of Texas, Austin, (1998).

[3] Kamaran F.,T.A.N,Al-Attar, H.Shareef, A.najib, G.Najeeb "PRIVATE-WIRELESS LOCAL AREA NETWORK (WLAN) FOR STUDENT FEEDBACK SYSTEM" *International Journal of Technical Research and Applications*. India, (2015).

[4] Mohit J. ,M. Danish, H.,Yadav, " CLOUD COMPUTING AND ONLINE OPERATING SYSTEM". *International Journal of Computer Technology and application*, India, 2011.

[5] Rajesh Sh., “Applications of Web based Operating System” *International Journal of Combined Research & Development (IJCRD),* eISSN:2321-225X;pISSN:2321-2241 Volume: 1; Issue: 2; June – (2013).

[6] Rajesh.S, P.Sharma, N.Kataria,"Web Operating System Impediments", *international Advanced Research Journal in Science, Engineering and Technology,* ISSN (Online) 2393-8021, AP, Dept. of Computer Science, Dev Samaj College for Women, Ferozepur City, India.(2015).

[7] Ananya Tripathi, “Web Operating Systems”, Amity School of Engineering and Technology, Amity University, Noida, India, (2014).

[8] Layla H, Emad A, *“*Assessing the quality of web sites”, Department of Management Information Systems, Zarqa Private University, Jordon , (2011).

[9] Daniel, Th."Fact Sheet Cloud Computing, Office of the Privacy Commissioner of Canada, May, (2016).

[10] Hesham.A,” Cloud Web-Based Operating System (Cloud WebOS)”, Department of Computer Science, Faculty of Information Technology, Applied Science University, Amman, Jordan, (2014).

[11] Henrik. S, Gustav, T." Web Operating System for Modern Smartphones", Master of Science Thesis, Chalmers University of Technology, Göteborg, Sweden, June,(2011).

[12] sanil .C.,S,*“***Trends in cloud operating system”**, *International Conference on Recent Trends in Information Technology and Computer Science,* Department of computer science, Gogate jogalekar collage,Ratangiri,(2011)

[13] Hui, S., C.,and Y., L., Pan,and C.,-H., Wu,and H., E., Yu,and K., Y., Cheng, and W., Huang, “Ezilla Fast, Deployment Cloud Toolkit with Network Infrastructure Service", *National Center for High-Performance Computing Taichung* ,Taiwan, (2012).

[14] Najib .K., A., and A.,T.,Al-Taani ,” Web Operating System and Computing on the web", Department of Computer Science, Yarmouk University of , Irbis, Jordan,(2005).

[15] Peter G. K.,*“*OVERVIEW OF THE WOS PROJECT*”*, Department d’informatique Universit´e Laval , Sainte-Foy(Quebec) Canada G1K 7P4, (2000).

[16] PETER KROPF,*”* OVERVIEW ABOUT THE RESOURCE SCHEDULING IN THE WEB OPERATING SYSTEM (WOS^{TM}), Universität Rostock, Canada, (1999).

[17] D.Bala, K., and ,R.A.M., Meshach,“Cloud Computing Along WebOS”, Pollachi **,**(2013)

[18] Jiaqi, D., P.Fasker, A.Fesak, and T.Stuart, " Benefits and Drawbacks of Cloud- Based Versus Traditionerp Systems **",** *Proceedings of the 2012-13course on Advanced Resource Planning W.J.H. van Groenendaal (ed.),* Tilburg University, Netherlands, (2012).

[19] A. Mufti, K. Salah ,” Web Operating System”, Department of Information and Computer Science King Fahd University(2002).

[20] Peter,.K., J.Plaice and H.Unger,"WebOS in Cloud Computing: A case study" Rostock, Germany, (1997).

[21] N.Thilak, Rajasekhar.B, “Android Operating System"**,** Department of Computer Science, ST. Mary’s College Of Engineering &Tech, (2011).

[22] A. M. Riad1, H., K. Elminir2, M., A., ElSoud3 and S.. F. Sabbeh4, *“*SEWOS: Bringing Semantics into Web operating System*”,* Damietta, Egypt, (2011).

[23] Amanda M., “Linux: The Operating System of the Cloud”, the Linux Foundation, (2009).

[24] Andreas F. K., Ricky S.D., "Cloud Computing a new approach for Hallstahammar’s IT companies"**,** *School of sustainable development & Technology, Master Thesis in IT Management,* (2010).

[25] Bill L., “Cloud Computing: What is Infrastructure as a Service"**,** TechNet Magazine,(2011).

[26] Fabio P., Peter B., Alessandro D,. Nico J. Thanos S., and Moritz Steiner, Alcatel-Lucent Bell Labs," **Toward a Cloud Operating System**" Service Infrastructure Research Dept. Computer Systems and Security ,Dept.(2010).{firstname.lastname}@alcatel-lucent.com URL http://is2.lse.ac.uk/asp/aspecis/20050111.pdf .

[27] F.Mountford, A.D.Phippen,*"*Feasibility Study into the use of Service Oriented architecture within the Atlantis University Portal", Network Research Group, section 2 – Information Systems Security & Web Technologies and Security ,University of Plymouth, Plymouth, United Kingdom, 2008. e-mail: info@cscan.org.

[28] Ankita.A., N.Chturvedi, D.Sisodia , "Supermarket Billing System" (Students of MCA IV- Semester, Group No.20),(2016).

[29] Jatinder M., *"* A Study of Factors Affecting Websites page Loading Speed for Efficient Web Performance", Department of Computer Science and IT, Bhaderwah Campus, University of Jammu, manhas.jatinder@gmail.com www.ijcaonline.org. ,(India), (2013).

[30] Alonso Miller,*"* Open Source vs. Proprietary Software*"*, CS639, (2011).

[31] Mark C.," rEd Hat EntErprisE Linux — tHE originaL cLoud opErating system*"*, Senior Director of Product Marketing Red Hat, (2012).

[32] Leon Sh. and R., Rosen., "Web Application Architecture Principles, Protocols and practices",(2009).

[33] Amin V., M., Dahlin, P., Eastham a, C., Yoshikawa Th., Anderson, and "WebOS: Software Support for Scalable Web Services"**, (**1997).

[34] Emmaunel B.,i, Oliver H., , M., Gunes, Thomas and C.S.,, Haw Hamburg, "Operating Systems for the IoT- Goals, Challenges, and Solutions" Saclay, France Germany, (2015).

[35] Datta D., "CLOUD OPERATING SYSTEM"**,** (2012).

[36] Sudhir , "Web Operating System **"** (2008).

[37] Haroon S., O.,"Client-Server Model", School of Computing Universiti Utara Malaysia Kedah, Malaysia, (2014**).**

[38] Steve W., "WHAT IS A WEB SERVER?" **(**1999**).**

[39] Lonnye B., Fardin K., Chris O., "APACHE WEB SERVER",(1999).

[40] Brandon F.,, "Web Browsers, Introduction to Web Design", (2011).

[41] Sung-W., W. , O., H. Alhazmi and Y., K. Malaiya, "Assessing Vulnerabilities in Apache and IIS HTTP Servers", (2011)

[42] A.Pascal, C.Asselin, "The Future of Cloud Computing", Site Point Pty Ltd, Collingwood, Victories, AU, (2013).

خلاصـــــــة

نظام تتشغيل الويب يعتبر مولد جديد لنظام التشغيل التقليدي ، يستخدم لخدمة وتطوير اطار بنموذج وشكل جديد عبر خدمة الانترنت بدلاً من الآلية الثابتة والمعتاد عليها اي (المحلية) المحددة ، هي خدمة الانترنت التي تمكن المستخدم الوصول بسهولة الى بياناته الخاصة باستخدام الاجهزة الذكية عن بعدٍ وفي اي مكان وزمان مجرد ان يكون الانترنت متوفراً. نظام تشغيل الويب (WebOS) نموذج جديد للحوسبة المعروفة بالـ (الحوسبة السحابية)والتوزيع العالمي الذي يتعامل مع نماذج الخدمة الخاصة بالحوسبة السحابي (IaaS, PaaS, SaaS) ، حيث ان نظام تشغيل الويب تقوم بتجهيزنا بالوظائف الاساسية المتعددة لنظام التشغيل التقليدي عبر متصفح الويب او عبر السحابة او الشبكة المحلية والعالمية بالاعتماد على استخدام تقنيات (ICT) . في الاطروحة المقترحة صممت نظام تشغيل الويب باستخدام بعض تكنولوجيا المعلومات مثال (X-cross platform – Apache- MySQL-PHP, Perl) كل هذه التكنولوجيا تواجدت في برنامج واحد والذي اعتمدنا عليه في بحثنا هذا ويدعى (XAMPP) وحققت في نوعين مختلفين من الشبكات وهما (الشبكة المحلية او الخاصة intranet –PrWebOS) و(الشبكة العالمية internet- PuWebOS).استخدام نوع من انواع هذه الشبكات المذكورة يتم من قبل المستخدم وهو الذي يقرر استخدام ايا منهما كان.والدخول الى نظام تشغيل الويب باستخدامه احد انواعه الذي اعتمدنا عليهما في بحثنا هذا وهما (EyeOS & Lucid) و اللذان صمما بـ (XAMPP) للشبكة المحلية وبالـ (LAMPP) للشبكة العالمية. في حالة الشبكة العالمية دلالة وجود خدمة الانترنت على جهاز الكومبيوتر لذا هنا يعتبر نظام تشغيل الويب هو عبارة عن كومبيوتر على الانترنت، لكن في الشبكة المحلية لاوجود لخدمة الانترنت لان في هذه الطريقة نعتمد على (WLAN) لذا هنا يعتبر نظام تشغيل الويب هو كومبيوتر على الانترانت . نظام تشغيل الويب بشكل متماثل يقوم بتقليد واجهة نظام التشغيل التقليدي وانها لاتعمل بشكل مباشر مع الموارد المادية المعنية لجهاز الكومبيوتر والمقصود به (hardware) وهنا نكمن بضرورة وجود النظام التشغيلي التقليدي للمستخدم لان وجوده مهم ولايستتى لنا ان نلغيه لان عن طريقه يتم تصميم نظام تشغيل الويب اي ان (نظام تشغيل الويب هو مكمل لنظام التشغيل التقليدي وليس بديلاً له).ان الفقرة الاكثر اهمية والتي جعلت نظام تشغيل الويب ان يكون اكثر شعبياً من نظام التشغيل التقليدي هي اسباب اقتصادية ومالية و تتلخص كالاتي(حيث تساعد على تقليل كلفة شراء والحاجة الى موارد و برامجيات جديدة). نظام تشغيل الويب تخدم المستخدم بخزن مجموعة هائلة من بياناته ومعلوماته، وبطريقة اساسية وواجهة بسيطة تسمح للمستخدم بالمشاركة تبادل الملفات، الاستماع الى الموسيقى ومشاهدة الفيديوهات، قراءة وكتابة النصوص ،والخ من الخدمات التي توفرها النظام الجديد المتطور مقابل اغناء المستخدمين بخدمة اكثر تطوراً واسرع تنفيذاً.

تمثيل نظام تشغيل الويب عبر الشبكة المحلية والعالمية

رسالة

مقدمة الى مجلس كلية العلوم

في جامعة السليمانية كجزء من متطلبات

نيل شهادة الماجستير في علوم

الحاسبات

من قبل

ســـارا عدنان أنور سلطان

بكالوريوس علوم حاسبات (٢٠٠٧)، جامعة كركوك

بإشراف

د. كامران حمه علي فرج

أستاذ مساعد

ذو الحجة ١٤٣٨ كانون الاول ٢٠١٦

پوخته

ویبه که به کار خستن سیستم (WebOS) نه وه ییکی نویێ ئاساییه به کار خستنی سیستم (TOS) ئه ویه را هات و ئاسان بکات په یکه رییك نوێ دەر بکه ویت پیرە دایم له لایان شوینی ئه نته رنیت له جیاتی بوون چاکی کرده وه بۆ ئامیر یکی دیار یکراو (ناوچه)ی ئەوە شۆینیکی ئینته رنیت نیته که به کار بەر ده توانیت بگات به بوی خۆ زانیاری به دووری هه رکوی له لایان هه ر کومبیوته ریك ئه و شوینه ی ئینته رنیت هه یه. (WebOS) شیوه ییکی نوییه هه ور کومپیوتینك – گلۆباللی بلاو ی کردەوه به تای به تی بھ کۆمپیوته ر که کار ده کات له سه ر سی خزمه تگوزاری مۆدیلی (هه ور کۆمپیوتینك) ژیرخان هەروەك خزمەت گوزارییك (IaaS) سەکو هەروەك خزمەتگوزارییك (SaaS) . سەرەرای ئەوەش ، (WebOS) چەندین کارییك دەسته بەر دەکات (TOS) بەسەر هەوری کۆمپیوتەر پشت ده بستن به له سەر پیداویستیەکانی زانیاریەکه پەیوەندی تیکنۆلەجیز (ICT). لەکه پیشنیار کەراو سیستم (WebOS) دروستی کرد لەلایان تەکنۆلوجیای ویب (XAMPP) ئەوە سەکۆییکی پەرینەوەیە ،(Apache) ، (MySQL) ،(PHP) و (Perl) ولیی کولیەوەزیاتر له دوو جوری جیاوازی (ICT) واته تایبەت نەتوۆرك (Private WebOS - intranet) وگشتی نەتوۆرك (Public WebOS- internet). ژمارەی بەکاربەر بریاری ئەدا بۆکارهینانی (PrWebOS) بی تەل ناوچەیی تۆری رووبەر (WLAN) یان (PuWebOS) لەلایانەکه دوو زور جوری باوی (WebOS) واته (EyeOS- Lucid) که تاقی دەکرینەوە (PuWebOS &PrWebOS) له ئون لاین " ئەنتەرنیت هەییه لەسەر کۆمپیوتەرەکه بۆیه (WebOS) کۆمپیوتەریکه لەسەر ئەنتەرنیتەکه" بەلام لەسەر بازیك ئەو "(WLAN) ئینیکی هەیه لەسەر کۆمپیوتەر لەسەر ئینترانیت " (WebOS) کتومته لاسایی کردنەوەی وبەستنەوی (TOS) ، تەنانەت (WebOS) پیکەوە راستەوخۆ رەفتار ناکات لەگەلەکه کۆمپیوتەری ئامیر ، بەلام کۆمپیوتەری خزمەتوەرگرهیشتاییت هه بیك (TOS) دەبیت .سەراری ئه وەش هۆکه لەپشت ناوبانگەکەی(WebOS) که ئەوە ئابووریه چونکه ئەو ئەوەیه که ئامیرەکه و بەرنامەکەم بکەوە بەگرنگی تی دەچیت (WebOS) ئەرك خەزن دەکەن زور بری گەورەی بەکاربەری زانیاری لەییك زور بەنەرەتی.

پیادە کردنی سەگۆی پەرینەوە بەکار هینان تۆری گشتی و تایبەت

نامەیەکە

پیشکەش کراوە بە ئە نجومەنی کۆلیژی زانست لە زانکۆی سلیمانی

وەك بە شیك لە پیداویستیەکانی بەدەست هینانی بروانەمەی ماستەر

لە زانستی کۆمپیوتەر

لەلایەن

سارا عەدنان ئەنوەر سولتان

بەکەلۆریۆس لە کۆمپیوتەر (٢٠٠٧)، زانکۆی کەرکووك

بە سەرپەرشتی

د. کامەران حەمە عەلی فەرەج

برۆفیسۆری یاریدەدەر

رە شەمێ ٢٧١٦ دیسەمبەر ٢٠١٦