

**AN APPLICATION BASED SOLUTION FOR MANAGING WEB SERVERS**

by

Elton DEMIRXHIU

**June 2012**

An Application Based Solution for Managing Web Servers

Copyright 2012 Elton Demirxhiu

To my parents

## **ABSTRACT**

An Application Based Solution for Managing Web Servers

Elton Demirxhiu

(June 2012)

Department of Computer Engineering

Epoka University, Tirana

Recently there has been a significant increase on the speed of internet and network connections worldwide, as new technologies are emerging and are acting on the market. Also cloud computing is becoming a key factor in IT area which is attracting serious investments in that field. Therefore, a better way of management and usage of resources of web hosts is needed in order to take the most benefit from web servers which are playing a key role in the evolution of cloud computing and internet.

This work presents a solution based on application of how to perform better management tasks on web hosts, how to make the necessary configurations of web servers in a much more easy way and how to get the most benefit from remote web servers by utilizing several features related to project management and development.

For this purpose, a Web based Server Management System that will perform all those required tasks, has been developed. The system is designed with the aspect of making it most user friendly interface where users can navigate and browse through different places in a familiar way.

## **ABSTRAKT**

Një Zgjidhje e Bazuar në Aplikacion për Menaxhimin e Serverave të Internetit

Elton Demirxhiu

(Qershor 2012)

Departamenti i Inxhinierisë Kompjuterike

Universiteti Epoka, Tiranë

Kohët e fundit, shpejtësia e internetit dhe e lidhjeve të rrjetit ka pësuar një rritje të konsiderueshme në mbarë botën, po ashtu edhe teknologjitë e reja po dalin në pah dhe po i bashkangjiten tregut. Gjithashtu “cloud computing” është duke u bërë një faktor kyç në fushën e IT-së e dhe është duke tërhequr mjaft interes në investime serioze për zhvillimin e kësaj fushë. Prandaj, nevojitet një mënyrë më efektive për menaxhimin dhe shfrytëzimin e serverave të internetit të cilët po luajnë një rol të rëndësishëm në këtë zhvillim të teknologjisë.

Kjo punë paraqet një mënyrë zgjidhje e bazuar në aplikacion se si mund të kemi një menaxhim më të mirë të serverave të internetit, si të bëjmë konfigurimet e duhura në to në një mënyrë sa më të lehtë dhe se si të marrim një përfitim sa më të madh nga resurset e serverave në distancë duke vënë në zbatim disa opsione të integruara që lidhen me menaxhimin dhe zhvillimin e projekteve software.

Për këtë qëllim, është ndërtuar një aplikacion për sistemin e menaxhimit të serverave të internetit, i cili do të kryejë të gjitha funksionet e kërkuara. Sistemi është i projektuar në mënyrë të atillë që të përmbajë një ndërfaqe të thjeshtëzuar, ku përdoruesit mund të lundrojnë lehtësisht në të.

## **ACKNOWLEDGEMENTS**

I would like to thank my supervisor, committee members, and my advisor, for their guidance and support throughout the course of this research.

Thanks also to my friends and colleagues and the department faculty and staff for making my time at Epoka University a great experience. I also want to extend my gratitude to the Information Systems Center of Epoka University, which provided me a lot of support, and gave me the chance to put my knowledge in practice.

Finally, thanks to my mother and father for their support and encouragement.

## TABLE OF CONTENTS

	Page
<b>DEDICATION</b> .....	iii
<b>ABSTRACT (English)</b> .....	iv
<b>ABSTRAKT (Shqip)</b> .....	v
<b>ACKNOWLEDGEMENTS</b> .....	vi
<b>TABLE OF CONTENTS</b> .....	vii
<b>1. INTRODUCTION</b> .....	1
<b>2. LITERATURE REVIEW</b> .....	2
2.1 Operating System Features .....	2
2.2 Cloud Computing .....	3
2.2.1 Types of Clouds .....	3
2.2.2 Cloud Services .....	4
2.3 Web Server.....	5
2.3.1 The hardware .....	6
2.3.2 The software .....	6
2.3.3 The client/server model.....	6
2.4 Web Hosting.....	7
2.4.1 Types of hosting.....	7
2.5 Multi-tier Applications on Internet.....	9
2.6 Web-based Enterprise Software .....	9
2.6.1 Hosted Application Advantages .....	9
2.7 Web Server Management Systems .....	10

	Page
<b>3. SOFTWARE ANALYSIS AND DESIGN .....</b>	<b>11</b>
3.1 Requirements .....	11
3.1.1 User-related Requirements .....	11
3.1.2 System Requirements .....	12
3.2 Structure Design .....	12
3.2.1 Main Interface .....	12
3.2.2 Component-based Development .....	13
3.2.3 Server Configuration .....	15
3.2.4 Account Management .....	17
3.2.5 Webmail .....	17
3.2.6 File Manager .....	18
3.2.7 Project Management and Development Tools .....	19
3.2.8 Other Plugins .....	21
3.3 Used Technologies .....	21
3.3.1 Apache HTTP Server .....	21
3.3.2 PHP .....	22
3.3.3 HTML5 .....	22
3.3.4 CSS3 .....	22
3.3.5 JavaScript .....	23
3.3.6 XML .....	23
3.3.7 jQuery .....	23
3.3.8 Ajax .....	24
3.3.9 JSON .....	25
<b>4. IMPLEMENTATION .....</b>	<b>26</b>
<b>5. CONCLUSIONS AND FUTURE WORK .....</b>	<b>31</b>
<b>REFERENCES .....</b>	<b>33</b>
<b>APPENDIX A .....</b>	<b>35</b>
<b>VITA .....</b>	<b>36</b>



## **1. INTRODUCTION**

Lots of effort has been put in to make sure that this system results into the product which is intended for, and it fits the user's needs in each and every aspect. The Web based Server Management System has been developed with a goal of benefitting both normal users and professionals. Users are able to access their contents and manage their work on their server and also performing the needed modifications or configurations easily from one single place, without having to install different applications on client or server in order to perform each task. Everything will be handled simply by using a web browser.

The first part of this thesis consists of important information related to the research part of the project. Here you can find general descriptions about web servers and the advantages of hosted enterprise applications on web.

The next part consists of analysis and design of the project including requirements, the design of the software structure with schematic explanations and brief descriptions explaining the main technologies used.

Finally, the last part contains the implementation, conclusions and future work related to the development and improvement of the project.

## 2. LITERATURE REVIEW

### 2.1 Operating System Features

An Operating System (OS) is a software program that manages the hardware and software resources of a computer. The OS performs basic tasks, such as controlling and allocating memory, prioritizing the processing of instructions, controlling input and output devices, facilitating networking, and managing files [1]. Some of the features of the operating systems are listed below: [2].

- *Software and hardware management* - The operating system is the bridge between computer hardware and software.
- *Constant API* - Application Program Interface (API) is a software that allows different applications that run on a computer to work on other computers also.
- *Execution of Programs* - Programs running in the computer and multitasking and multithreading features are completely dependent on the OS.
- *Interruptions* - The operating system allows and handles different interrupts.
- *Managing Memory* - The operating system provides the memory for the programs that are executed at any moment.
- *Networking* - The operating system makes it possible for one computer to communicate with other computers in a network.
- *Security* - An operating system in the computer takes care of all security issues of computer and data in it.

## 2.2 Cloud Computing

Cloud computing is a term that describes the means of delivering any and all Information Technology - from computing power to computing infrastructure, applications, business processes and personal collaboration - to end users as a service wherever and whenever they need it. The cloud in cloud computing is the set of hardware, software, networks, storage, services, and interfaces that combine to deliver aspects of computing as a service. Shared resources, software and information are provided to computers and other devices on demand. It allows people to do things they want to do on a computer without the need for them to buy and build an IT infrastructure or to understand the underlying technology. [3]

### 2.2.1 Types of Clouds

There are different kinds of cloud deployment models which provide different kinds of services and are classified into four types: [4]

- *Public Cloud* - The computing infrastructure is shared between any organizations.
- *Private Cloud* - The computing infrastructure is dedicated to a particular organization and not shared with other organizations. Private clouds are more expensive and more secure when compared to public clouds.
- *Hybrid Cloud* - Organizations may host critical applications on private clouds and applications with relatively less security concerns on the public cloud. The usage of both private and public clouds together is called hybrid cloud.
- *Community Cloud* - Community cloud involves sharing of computing infrastructure in between organizations of the same community.

## 2.2.2 Cloud Services

Cloud computing providers offer their services according to three fundamental models (Figure 1): [5]

- *Infrastructure as a Service (IaaS)* - The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications.

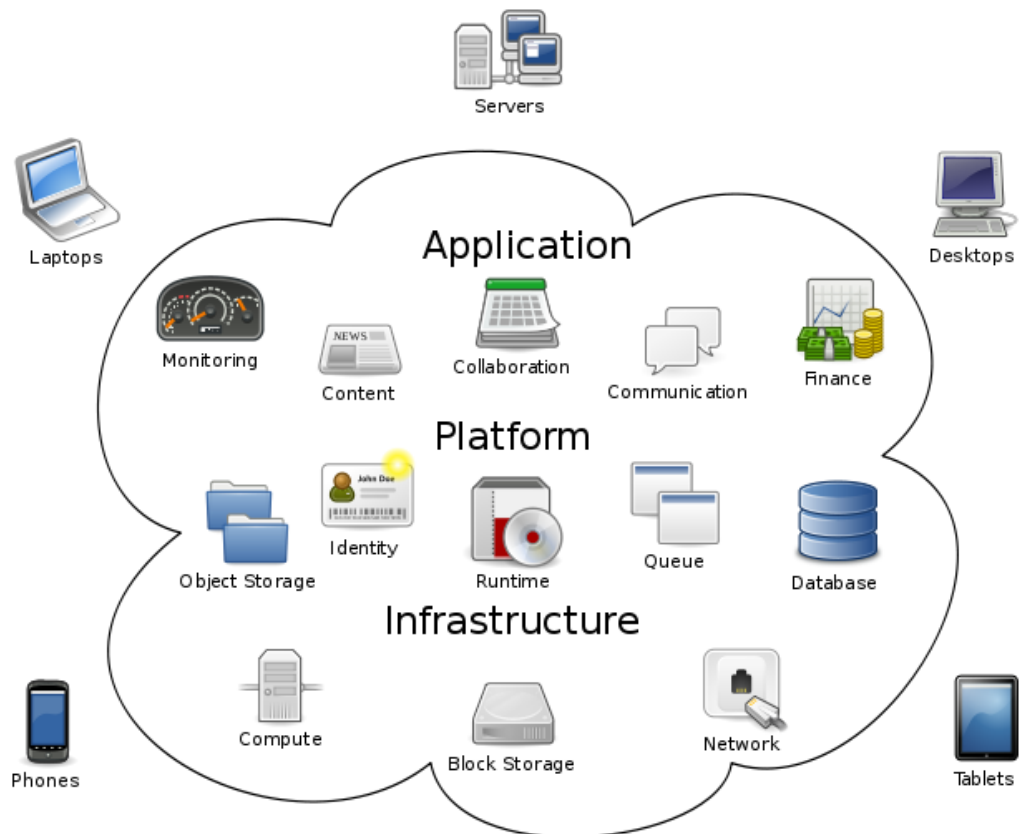


Figure 1. Cloud Computing Services

- *Platform as a Service (PaaS)* - The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.
- *Software as a Service (SaaS)* - The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser, or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

### **2.3 Web Server**

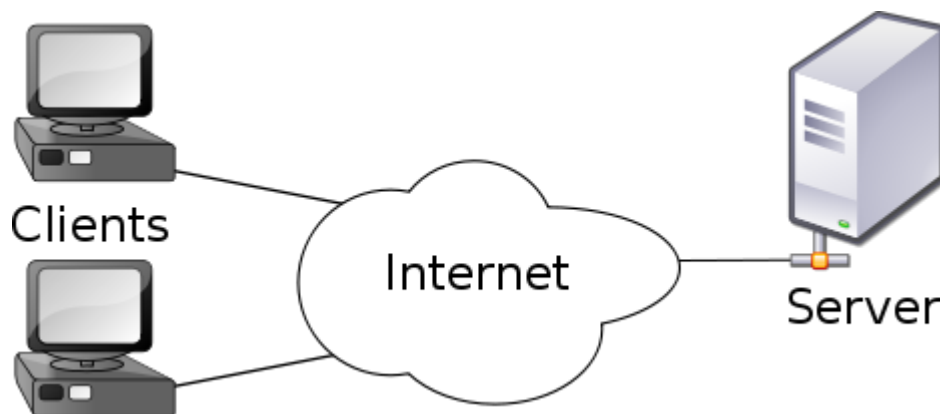
Web server can refer to either the hardware (the computer) or the software (the computer application) that helps to deliver Web content that can be accessed through the Internet. The most common use of web servers is to host websites, but there are other uses such as data storage or running enterprise applications. [6]

### ***2.3.1 The web server computer - the hardware***

A web site is a collection of web pages which are digital files generally written using HTML. For a web site to be available to everyone in the world at all times, it need to be stored or "hosted" on a computer that is connected to the internet. Such a computer is known as a Web Server.

### ***2.3.2 The web server computer - the software***

A web server program is software that runs on the web site hosting Server computer. Its main purpose is serving web pages; which mean it waits for requests from web browsers (clients) and responds by sending the required data back.



**Figure 2.** Client Server Model

---

### ***2.3.3 The client/server model***

The client/server model is a computing model that acts as distributed application which partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients. [7] A server machine is a host that is running one or more server programs which share their resources with clients. A client does not share any of its resources, but requests a server's content or service.

## 2.4 Web Hosting

A web hosting service is a type of Internet hosting service that allows individuals and organizations to make their website accessible via the World Wide Web. Web hosts are companies that provide space on a server owned or leased for use by clients, as well as providing Internet connectivity, typically in a data center. The most basic feature is web page hosting, where files can be uploaded via File Transfer Protocol (FTP) or a Web interface.

### 2.4.1 Types of hosting

Internet hosting services can run Web servers. The company may use the computer as a website host to provide details of their goods and services and serve for online orders. [8]

- *Free web hosting service*: offered by different companies with limited services, sometimes supported by advertisements.
- *Shared web hosting service*: one's website is placed on the same server as many other sites, ranging from a few to hundreds or thousands. Typically, all domains may share a common pool of server resources, such as RAM and the CPU.
- *Reseller web hosting*: allows clients to become web hosts themselves. Resellers provide a nearly identical service to their provider's shared hosting plan and provide the technical support themselves.
- *Virtual Dedicated Server*: also known as a Virtual Private Server (VPS), divides server resources into virtual servers, where resources can be allocated in a way that does not directly reflect the underlying hardware. The users may

have root access to their own virtual space. Customers are sometimes responsible for patching and maintaining the server.

- *Dedicated hosting service*: the user gets his or her own Web server and gains full control over it (user has root); however, the user typically does not own the server. The user has full administrative access to the server, which means the client is responsible for the security and maintenance of his own dedicated server.
- *Managed hosting service*: the user gets his or her own Web server but is not allowed full control over it (user is denied root access); however, they are allowed to manage their data via FTP or other remote management tools. The user typically does not own the server. The server is leased to the client.
- *Cloud hosting*: is a new type of hosting platform that allows customers powerful, scalable and reliable hosting based on clustered load-balanced servers and utility billing. A cloud hosted website may be more reliable than alternatives since other computers in the cloud can compensate when a single piece of hardware goes down. On the other case the lack of centralization may give users less control on where their data is located which could be a problem for users with data security or privacy concerns.
- *Clustered hosting*: having multiple servers hosting the same content for better resource utilization. Clustered Servers are a perfect solution for high-availability dedicated hosting, or creating a scalable web hosting solution. A cluster may separate web serving from database hosting capability.
- *Grid hosting*: this form of distributed hosting is when a server cluster acts like a grid and is composed of multiple nodes.



- *Home server*: usually a single machine placed in a private residence can be used to host web sites. Users must have static IP address from ISPs.

## **2.5 Multi-tier Applications on Internet**

Modern Internet applications are designed using multiple tiers. A multi-tier architecture provides a flexible, modular approach for designing such applications. Each application tier provides a particular functionality to its preceding tier and uses the functionality provided by its successor to carry out its part of the overall request processing. Request processing at each tier consists of an interleaving of periods where the tier performs some work on the request and periods where it awaits some service it requested from its successor tier. [9]

## **2.6 Web-based Enterprise Software**

Enterprise software provides functionality that supports and enables core and support processes in an organization, such as planning, financial management, team collaboration, or customer relationship management. A main fact to notice is that enterprise software is sold to companies, not to individuals.

### ***2.6.1 Hosted Application Advantages***

- *Flexibility and speed of deployment*: Since no local installation is necessary for a hosted application, users can start to use an application very quickly.
- *No infrastructure investments*: The customer doesn't have to make an investment in server hardware. The only necessary things needed are client PCs with web browsers and an Internet connection.

- *Server platform independence*: Vendors do not have to develop for different server platforms, but can select the platform they feel most comfortable with.
- *Cost reduction for installation and maintenance*: There is no cost for the installation and maintenance of proprietary client software on users' computers except web browsers.

## **2.7 Web Server Management Systems**

Web server management software is software designed to run on web servers to manage all of its aspects via a web interface. These software types are often referred to as a "control panel" as they offer the interface panel to control the server. These types of control panel solutions are usually used by web hosting companies and web server administrators. A control panel, in web hosting, is a web-based interface provided by the hosting company that allows customers to manage their various hosted services in a single place. A full structured table with detailed information about current operating web server management software worldwide can be found on Appendix A.

### **3. SOFTWARE ANALYSIS AND DESIGN**

#### **3.1 Requirements**

The application consists of several features including: User Management, Server Configuration, Account Administration, System Monitor, File Manager, Code Editor, Image Editor, Database Manager, Project Management Tools, etc. It will be used by Global Administrator, System Administrators, Developers and Web Designers, Non-technical and Specific Users.

##### ***3.1.1 User-related Requirements***

A central, role-based user authentication will manage access permissions to the system. User recognition will be performed by using Sessions and HTTP cookies. They store a user ID and a session ID that will be compared with the entries in the database. Users can have access if their user account is stored in the database. Furthermore, users may only execute a task or view an item of the application if they have sufficient rights. A user account can be activated or deactivated. Necessary user information are: Login name, Password, Real name, Email address, Access rights. A login procedure generates a session and a cookie for the client's browser. A logout procedure deletes the session and the cookie. Each session is time-dependent and contains an expiry date, which ensures that each session is unique and cannot be used more than once. Before a user can execute a specific task it will be ensured that he owns access rights to do this. The elements, in which the users have no access, will not be displayed.

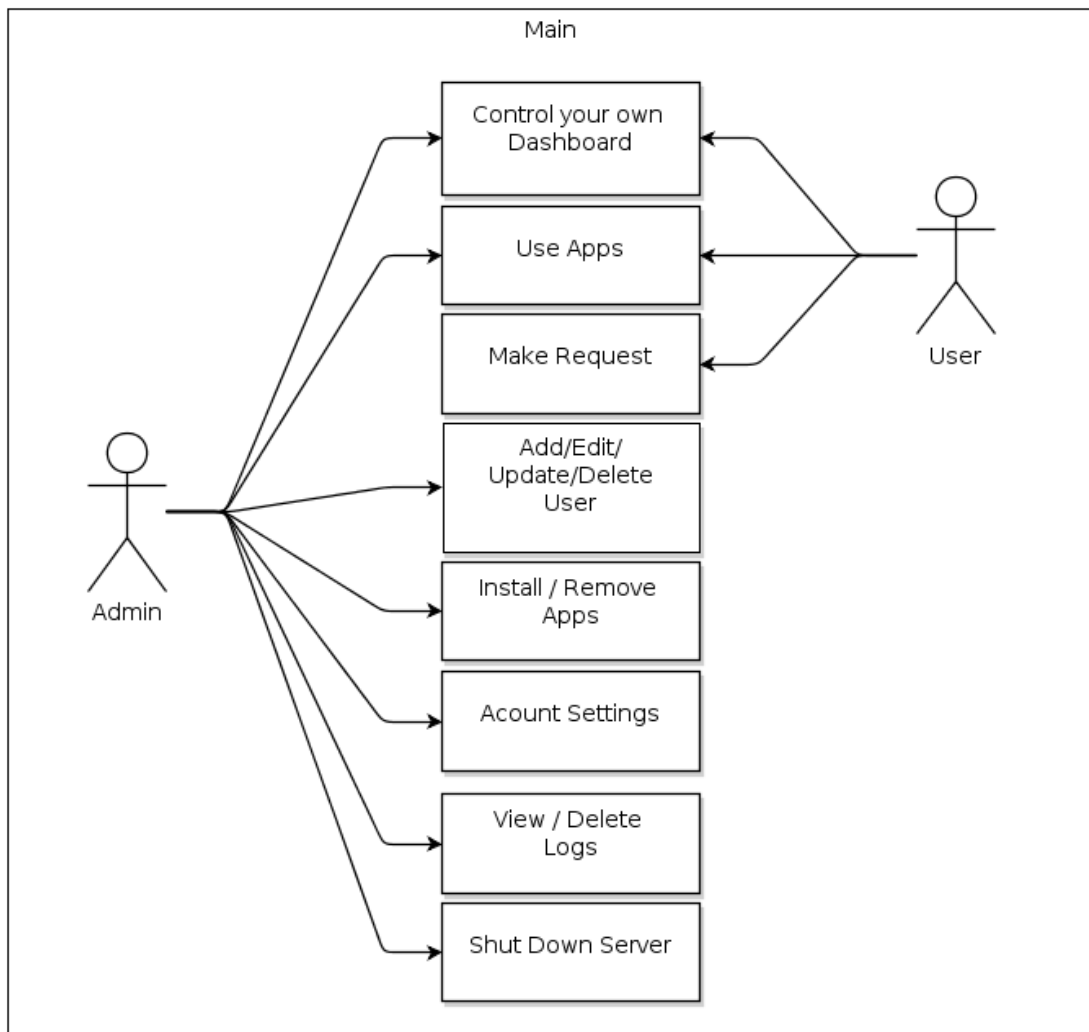
### ***3.1.2 System Requirements***

The application will run on Debian based Linux Operating Systems. Before installation, it is required that apache php5 and mysql are previously installed on server and are running properly. After the system has been deployed on the web server, the installation process will continue and the installation steps will be followed from the user interface on your browser. During the installation, the wizard will evaluate the components that consist within the main application. If the owner of the host who is installing the application, doesn't have the privileges required by the component to work properly, that specific component will not be installed. (e.g. If the user is prohibited from using terminal, the terminal app won't be installed)

## **3.2 Structure Design**

### ***3.2.1 Main Interface***

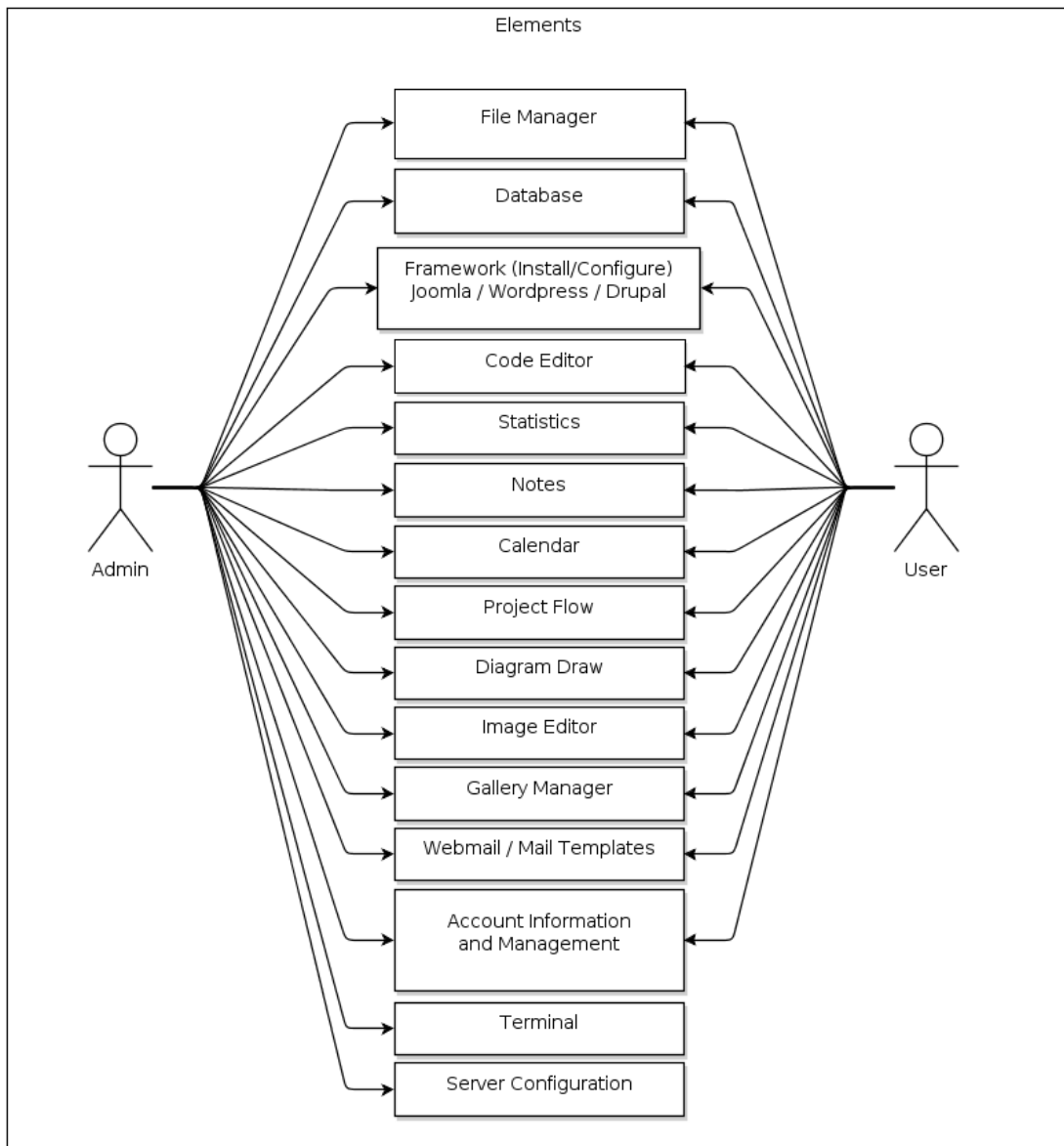
The main interface of the application contains the dashboard which can be organized according to user preferences. Normal users can make request to their administrators for specific needs and administrator can accept or deny their request. Normal users can have access only to the authorized parts, while administrators grant full access everywhere. Administrator can add or remove a user, install or remove apps, edit account settings, view server logs etc. If the administrator is controlling his home or office server, he may even shut it down or reboot. (Figure 3)



**Figure 3.** The main interface of the application

### ***3.2.2 Component –based development***

With component-based development it is possible to divide large and complex software systems into smaller, less complex modules. Since these modules have an encapsulated function they can be decoupled from each other and thus be implemented in parallel by different developers, regardless of location, independently of each other's work. Thus development time is reduced.



**Figure 4.** The components of the application

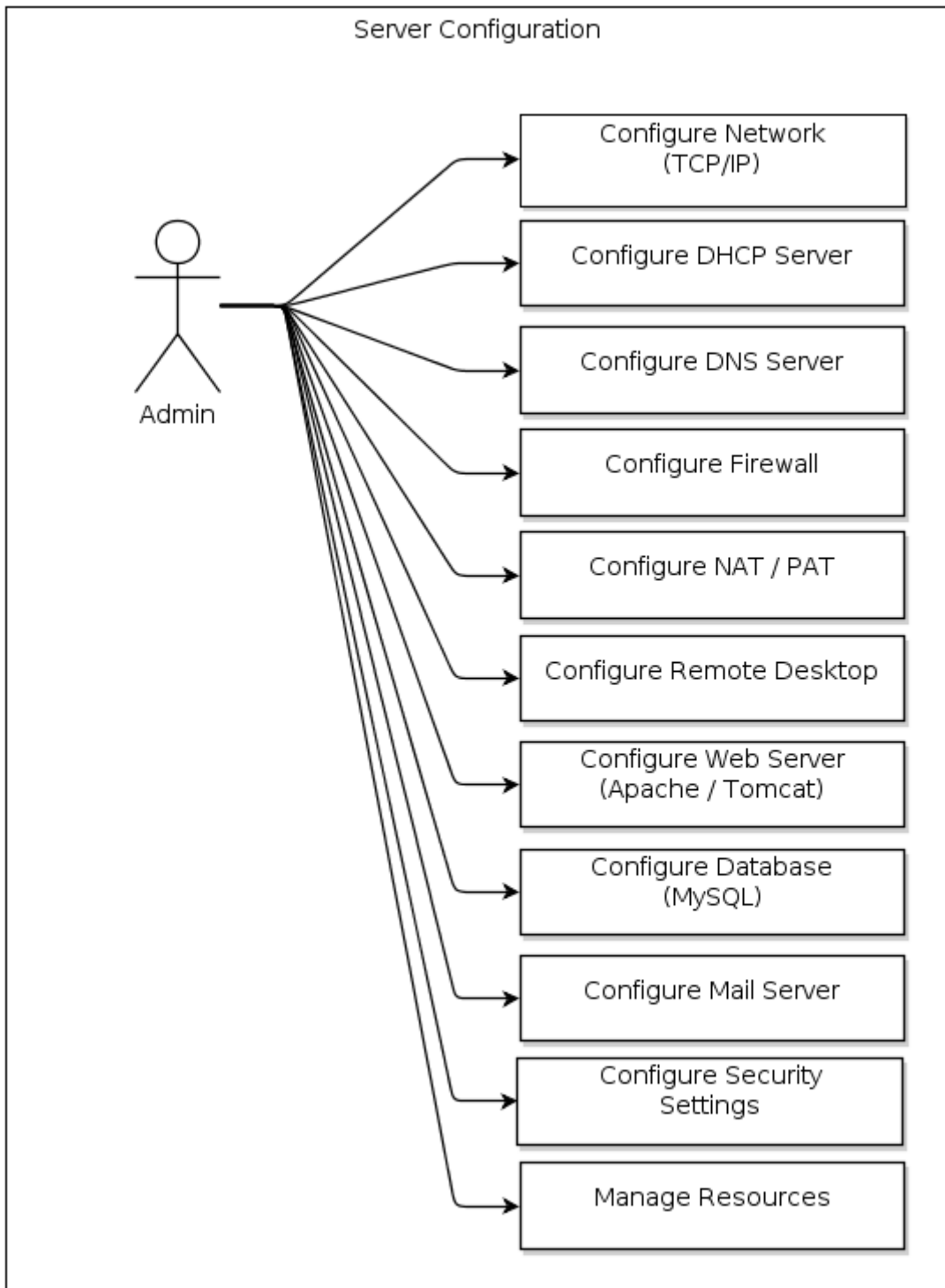
Moreover component-based technology promises higher quality because the specific components can be tested individually with unit tests and the component assembly has to be tested as well with integration tests. Another advantage is that software systems which consist of several components are more flexible than monolithic software systems. If a software system has to fulfill new requirements which were

not anticipated, it is easier to recompose the components than to modify a monolithic system. Component compositions are often easier to maintain because each component can be maintained independently [10]. Our application is a component-based one. Alongside with the main interface, it contains several components as described in (Figure 4).

### ***3.2.3 Server Configuration***

The server configurations can be performed only by administrator. From server configuration panel administrator can make the configurations about the network settings, web server (Apache), database server, mail server, demote desktop settings. There are also options to specify the usage of resources for different clients. By using Virtual Host, administrator can configure the server to have multiple domain names on a single IP.

If you want to configure the server in order to function only as a network device you may exclude other features and apps and keep only the configuration panel at the time of install. In that case the only needed elements are: Network Configuration, Firewall Configuration, DNS and DHCP Server, Network Address Translation, Port Address Translation and Security Settings. (Figure 5)

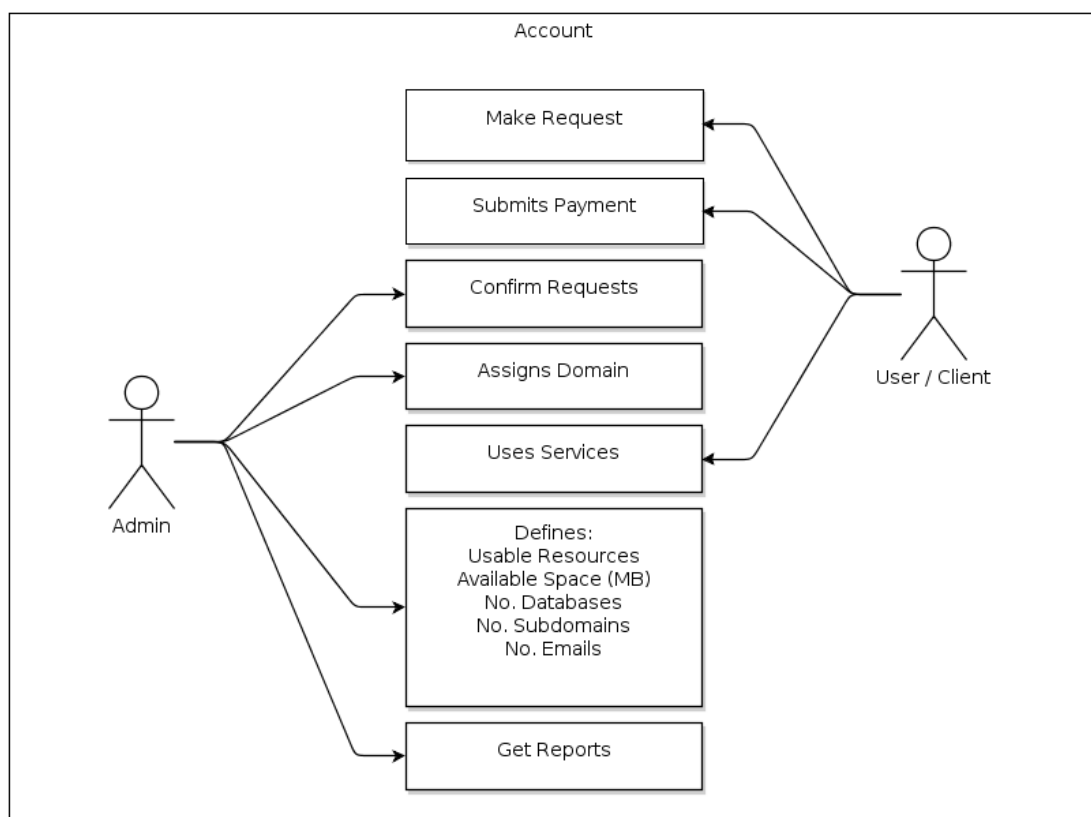


**Figure 5.** Server Configuration Elements



### 3.2.4 Account Management

This feature is available only in the case of web host administration. Administrators can assign domains and they can define the host packages available to users. Users can extend their subscription by submitting payments in order to use the host services. They can make requests to host administrator for different changes in their host plan. Administrator can accept or deny user requests and they also can get reports on current and existing users on the system. (Figure 6)



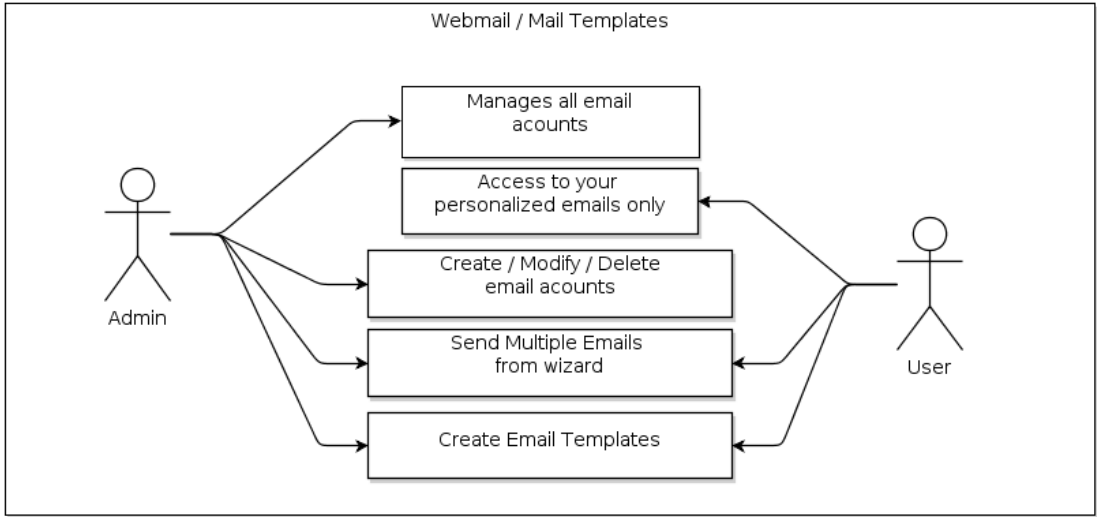
**Figure 6.** Account Administration Schema

---

### 3.2.5 Webmail

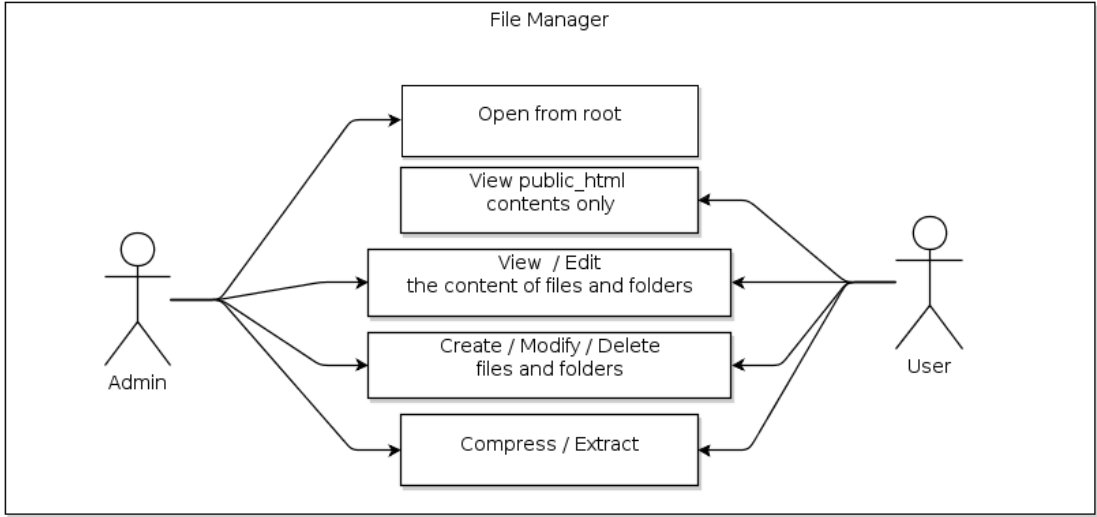
This feature includes the email management, and also enables users to send multiple emails from templates in the name of their company. Administrator can manage all email accounts. Usually the email accounts come up with the company

domain name extension. Administrator has the right to add or delete new email accounts and he can also switch to global email applications like Gmail or Windows Live or he may choose use a new webmail application. Users have only access to their personal emails. (Figure 7)



**Figure 7.** Webmail

**3.2.6 File Manager**



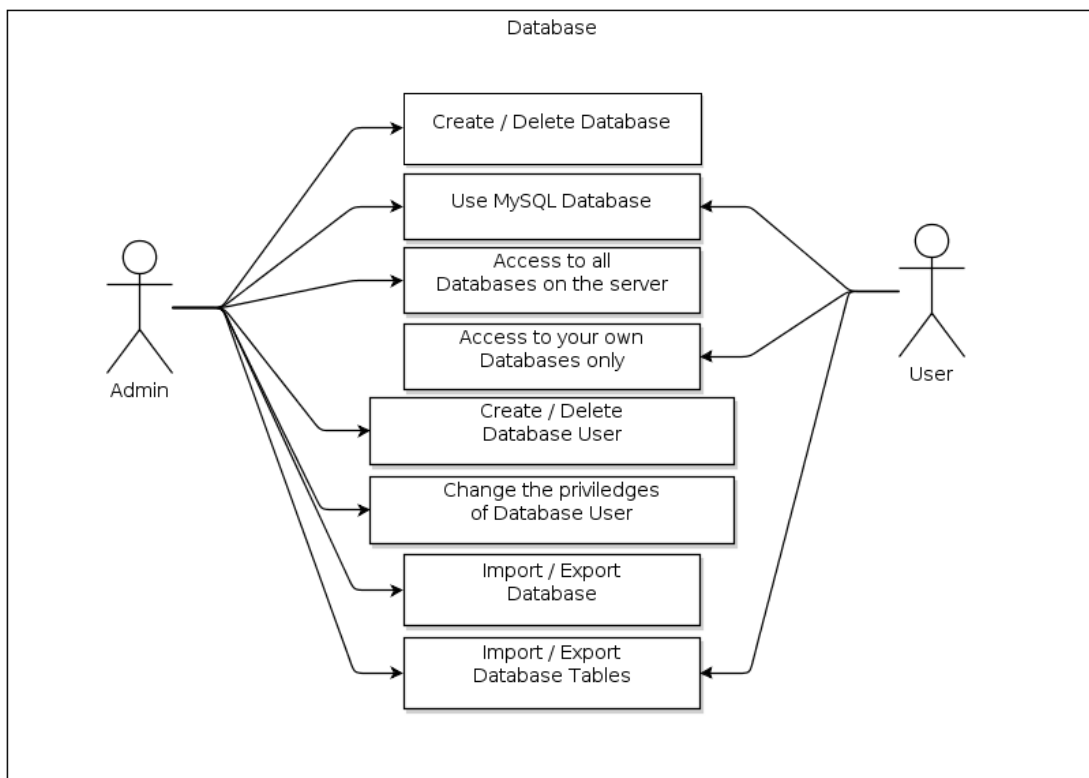
**Figure 8.** File Manager

A web based file management system that will enable users navigate through files and folders on serve and to upload or download files to/from their it. (Figure 8)

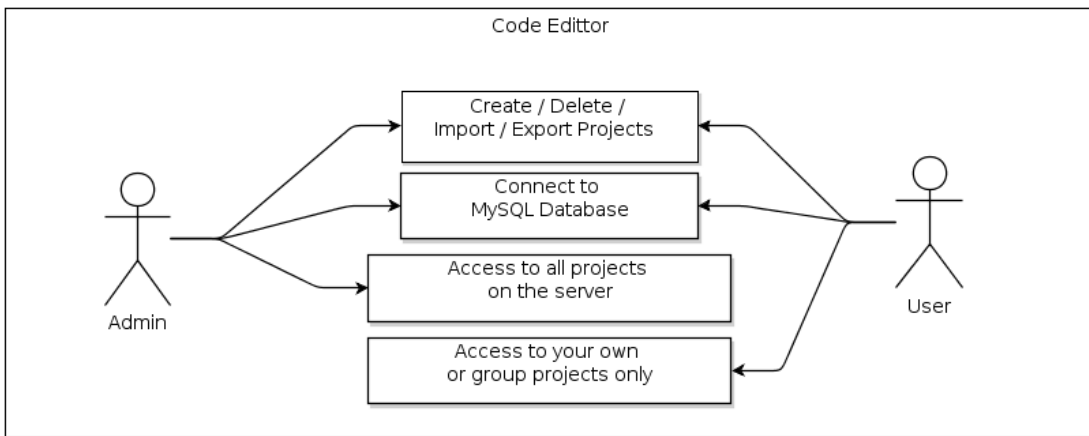
### 3.2.7 Project Management and Development Tools

Software development needs highly capable communication systems. Usually there are several developers engaged in a software project. In times of globalization it is a common practice to collaborate internationally. That means that world-wide distributed project teams must notify each other about the work progress. So we have included some of those features as a part of our project.

Project development apps include Code Editor, which is a web based IDE for creating PHP projects (Figure 10) and phpMyAdmin as database management system for managing MySQL databases (Figure 9).



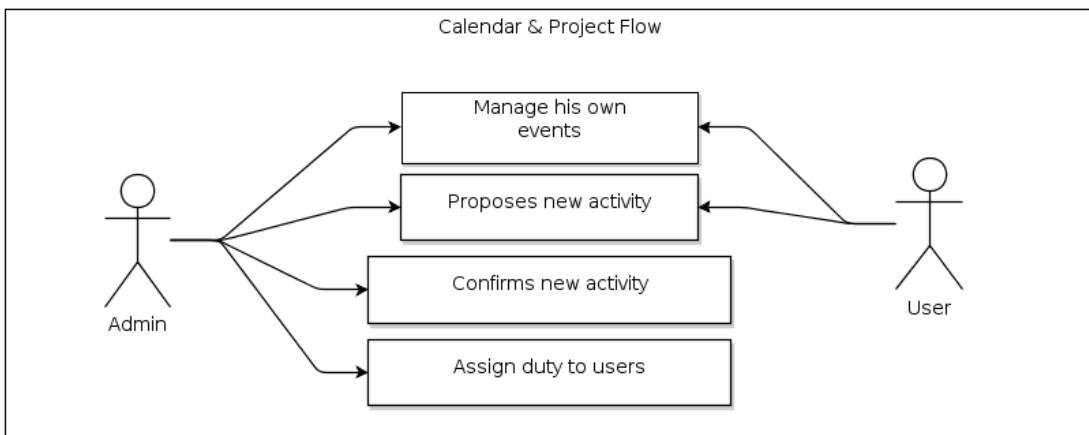
**Figure 9.** Database Management



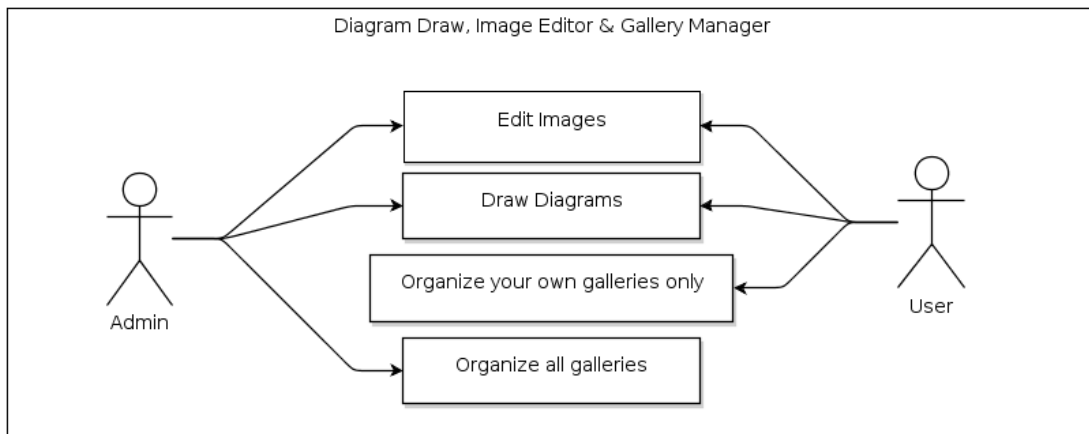
**Figure 10.** Code Edittor

Managers can use project management tools for creating the project flow diagrams and making them visible to the developers. They may also assign new tasks related to the project which are going to be shown on users' calendar. (Figure 11)

There are a lot of advantages of such a group of applications. Users can work on their projects from any place simply by having only internet connection.



**Figure 11.** Project Management and Calendar



**Figure 12.** Media and Gallery Elements

### ***3.2.8 Other Plugins***

The other list of elements composing the project consists of System Monitoring and Statistics from one side and Gallery Manager and Image Editor (Figure 12) from the other side. Users may view their web site traffic from Statistics. By using Image Editor or Gallery Manager they may edit and organize images online on their server without the need of downloading them on client PC and upload them again.

## **3.3 Used Technologies**

### ***3.3.1 Apache HTTP Server***

The Apache HTTP Server is web server software mainly used to run php and perl scripts. It has played a key role in the initial growth of the World Wide Web.

The application mainly runs on Unix based Operating Systems but it is also available for other platforms. Released under the Apache License, Apache is open-source software and it is developed and maintained by an open community of developers.

### ***3.3.2 PHP***

PHP is a general-purpose server-side scripting language originally designed for Web development to produce dynamic Web pages. It is one of the first developed server-side scripting languages to be embedded into an HTML source document, rather than calling an external file to process data. PHP code is interpreted by a Web server with a PHP processor. It also supports Object Oriented features. PHP is free software released under the PHP License.

### ***3.3.3 HTML5***

HTML5 is a markup language for structuring and presenting content for the World Wide Web, and is a core technology of the Internet originally proposed by Opera Software. It is the fifth revision of the HTML standard and it is still under development. In particular, HTML5 adds many new syntactical features. These include the new multimedia handling elements, as well as the integration of scalable vector graphics (SVG) content. New attributes or elements have been introduced, while some others have been removed, changed, redefined or standardized. An HTML5 browser will be flexible in handling incorrect syntax. HTML5 is designed so that old browsers can safely ignore new HTML5 constructs.

### ***3.3.4 CSS3***

CSS 3 is introduced in order to have the best combination with HTML5. It is still under development. Unlike CSS 2, which is a large single specification defining various features, CSS 3 is divided into several separate documents called "modules".

Each module adds new capabilities or extends features defined in CSS 2, over preserving backward compatibility. CSS 3 has introduced many new features and attributes. The aim is to use the standardized attributes in cross-browser by achieving the same result.

### ***3.3.5 JavaScript***

JavaScript is a client side scripting language supported by almost all browsers. It can interact with HTML source code, enhancing the functionality and appearance of the content of web pages. It has a C - like syntax and despite the name "JavaScript", it is independent from Java.

### ***3.3.6 XML***

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The design goals of XML emphasize simplicity, generality, and usability over the Internet. It is a textual data format with strong support via Unicode for the languages of the world. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures, for example in web services. Many application programming interfaces (APIs) have been developed for software developers in order to process XML data, and read XML-based languages.

### ***3.3.7 jQuery***

jQuery is a cross-browser JavaScript library designed to simplify the client-side scripting of HTML. jQuery is the most popular JavaScript library in use today.

jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and web applications. Because of jQuery's architecture, other developers can use its constructs to create plug-in code to extend its functionality. Currently there are thousands of jQuery plug-ins available on the web that cover a wide range of functionality such as Ajax helpers, webservices, datagrids, dynamic lists, XML tools, drag and drop, events, cookie handling, modal windows etc. jQuery is free, open source software, dual-licensed under the MIT License or the GNU General Public License.

### ***3.3.8 Ajax***

Ajax (Asynchronous JavaScript and XML) is a group of interrelated web development techniques used on the client-side to create asynchronous web applications. With Ajax, web applications can send data to, and retrieve data from, a server asynchronously (in the background) without interfering with the display and behavior of the existing page. Data is usually retrieved using the XMLHttpRequest object. Despite the name, the use of XML is not required (JSON is often used instead), and the requests do not need to be asynchronous. Ajax is not a single technology, but a group of technologies. HTML and CSS can be used in combination to mark up and style information. The DOM is accessed with JavaScript to dynamically display, and to allow the user to interact with the information presented. JavaScript and the XMLHttpRequest object provide a method for exchanging data asynchronously between browser and server to avoid full page reloads.



### **3.3.9 JSON**

JSON, or JavaScript Object Notation, is a lightweight text-based open standard designed for human-readable data interchange. It is derived from the JavaScript scripting language for representing simple data structures and associative arrays, called objects. Despite its relationship to JavaScript, it is language-independent, with parsers available for many languages. The JSON format is often used for serializing and transmitting structured data over a network connection. It is used primarily to transmit data between a server and web application, serving as an alternative to XML. JSON is often used in Ajax techniques. Ajax commonly used XML as the data interchange format but many developers have also used JSON to pass Ajax updates between the server and the client.

## 4. IMPLEMENTATION

The development of a successful web application involves many different kinds of design, including functional design, software architecture, business process or workflow design, user interface design, and database design. More than any other type of application, web applications require effective integrated use of software engineering practices and interaction design principles.

Nowadays, the design and development of such web applications have evolved very substantially. Today's sites include JavaScript for client-side, dynamically generated web pages, and a complex "back-end" that often includes extensive processing logic written in server side language, a database management system, and specialized components for security.

Applications developed for high-volume use on the World Wide Web are highly complex, and illustrate the practical application of principles and guidelines for human-computer interaction. The best designed web applications are characterized by high reliability, high availability, high security, and rapid interactive response. In summary, there is a significant challenge to unify design principles from the various disciplines that influence the development of web applications and their subsequent usefulness.

The system is designed with the aspect of making it most user friendly interface where users can navigate and browse through different places with ease. The GUI is implemented using HTML5, CSS3, JavaScript and jQuery which works on almost every internet browser available today. The backend part is done using PHP and Object Oriented PHP.

Different from the classic organization of control panels where links and options are located on the left side of the page, my application contains a menu and a simple sliding ribbon located on the top. The ribbon keeps the users favorite apps that are pinned here and the opened apps which are actually in use making it easier to switch between apps. The ribbon is built in a way that it slides by when the list of icons on it exceed the ribbon's width. This is achieved by using jQuery mouse events and slide effects. The ribbon can be hidden or shown depending on the users screen preferences.

The application has two main interfaces. Both interfaces are similar in design but they differ in construction. The first one is designed in order to be able to scroll the list of sections defined in HTML5 with their content inside and it is called "page mode". This interface is used to show pages like users' dashboard, the list of apps and system monitor or statistics. The second one is the interface to show the apps which are linked with modules. Beside the first interface, it consists of an iframe which shows the content of the apps in it. It calculates its size by using jQuery window event and operates in 100% width and height mode. It also excludes the window scroller and uses only the scroller of the iframe. This is called "app mode" and is used to show apps like File Manager, phpMyAdmin, Code Editor and Terminal. The main advantage of it is that the CSS of apps is not affected by the main CSS.

First time the user logs in, the first page shown is not the dashboard like other classic control panel, but the list of apps making it easier for users to navigate through apps especially users from tablets. Anyway users may switch to the dashboard and organize it whenever they want.

The application is designed to handle more than one opened app and keeping it at the position or state where the user was working in. The content of a minimized app is opened but is being hidden. When the user wants to access it, the current app is being hidden and the app in minimized state is shown. It can be achieved by using jQuery hide and show events. That feature is useful but it makes a significant load to the client computer so it will be enabled or disabled from panel settings. When this feature is disabled, switching between apps is done by closing and opening them again. Each app or feature has its own icon. Icons listed in the app list have the size 64x64 pixels while icons listed on the ribbon are 32x32 pixels.

Some parts of the application mainly related to different apps are implemented by using the base of open source. All apps are placed in a folder named “modules” and the contents of apps are organized in separate folders inside that folder which are specified by app name. In modules it is created a php file for each app which specifies the iframe and the source of index. These php files have the name of the app and they are included in app section in the main interface. The link is set according to the app name and it also gets the icon which has the same name with the app. It points to the index page and specifies the mode and the app name and according to the name we get from url, we define the path of the app. Basically we call the php file from the modules folder. Another thing to mention is that we have to configure the configuration file of the apps which are ready in order to be compatible with the system, as in the case of phpMyAdmin for MySQL database management, Shellinabox, for linux terminal and elfinder for file manager. Other apps are built by combining plugins with our implementation and forming a working app to solve the specific task. For example Code Editor is built on the base of codemirror which uses

a textarea for code editing and code suggestions. It also consists of a tree view to display files and folders on the left and tabs on the upper part in order to switch between opened files. Code mirror is also used to edit the configuration files of linux server. This application consists also of several plugins including text editor, graph plotter, calendar etc. Those plugins are located inside plugins folder and can be included everywhere in the application without the need of using iframe. Statistics and System Monitor use flot for graph plotting. It takes some values from files or from database as input and then display the result in a graph.

The open source references which are integrated within the project include: [11]

- *elFinder* – file manager for web [12]
- *phpMyAdmin* – handles the administration of MySQL for web [13]
- *CodeMirror* – In-browser code editing [14]
- *Jwysiwyg* – Web based text editor [15]
- *Shellinabox* – Web based Ajax terminal [16]
- *FullCalendar* – Web based Ajax calendar [17]
- *Flot* – Graph plotting for jQuery [18]
- Other plugins like easing plugins and resizing events. [19]

Later, when we will introduce the install and uninstall features, we will use JSON for parsing elements and defining the paths.

The implementation of the application is provided with a CD at the end.

The screenshots in (Figure13 and Figure 14) represent the view of the project from web browser.

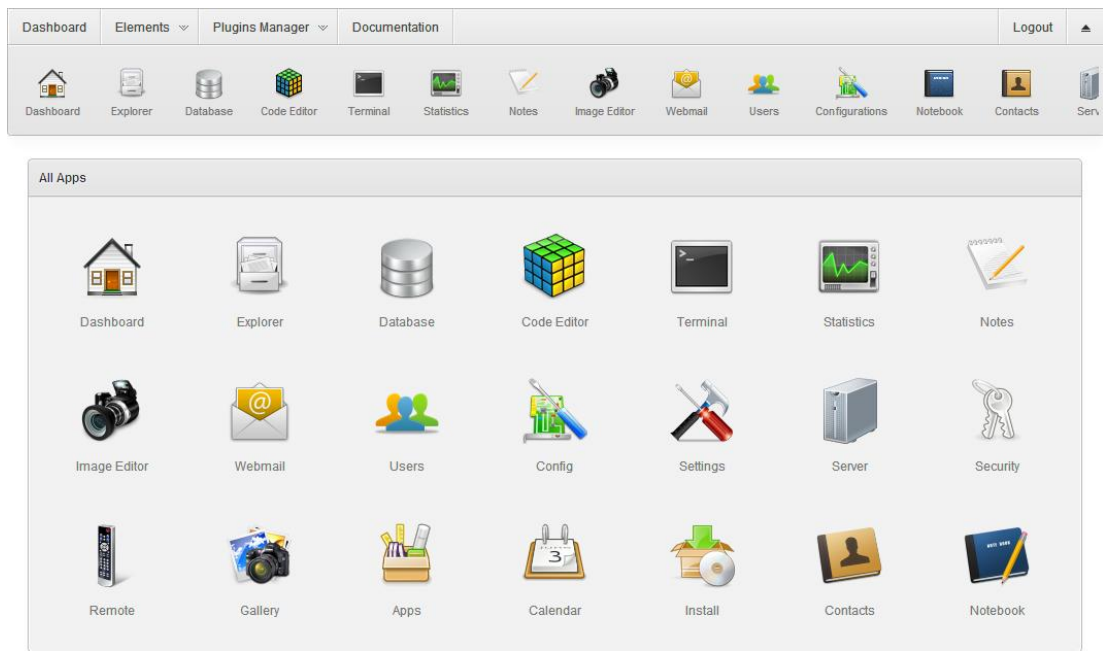


Figure 13. Screenshot from application list view of the project

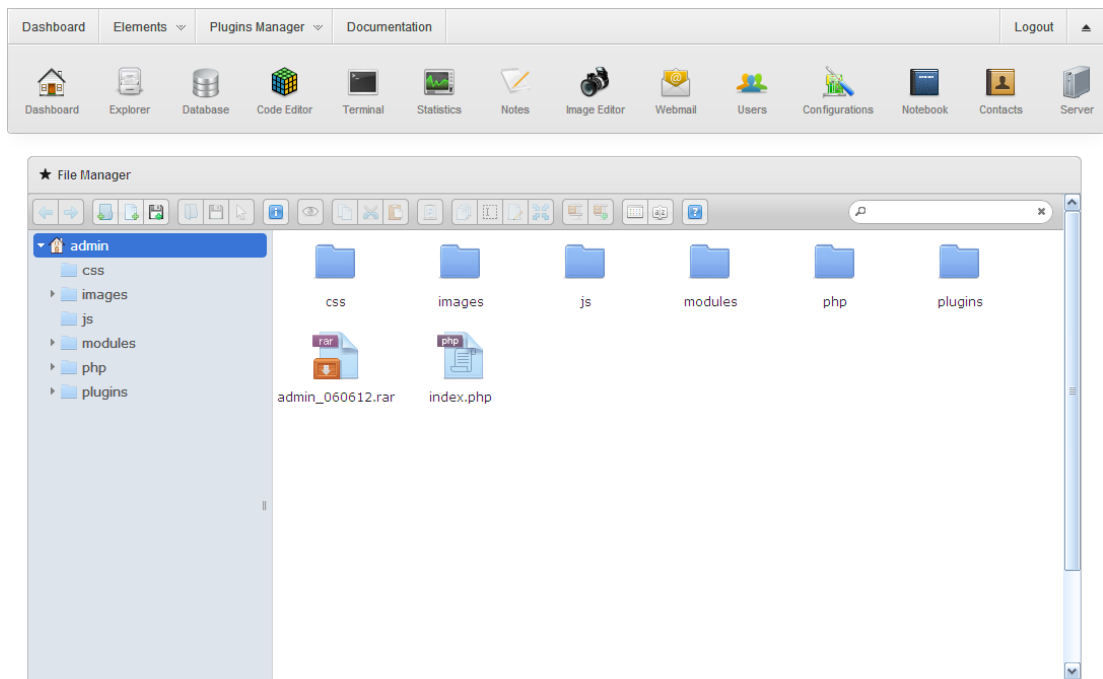


Figure 14. Screenshot of the file manager of the project

## 5. CONCLUSIONS AND FUTURE WORK

We do expect that our system would be a significant contribution to the field of computer science in the area of the management of servers and hosts. By using it, users would not be limited to do only simple management tasks. Of course they may carry almost every management and configuration task related to their server or host, but they are also provided with a wide area of several complete features including File Manager, Project Management Tools, Code Editor, Database Manager, etc. With the increasing speed of internet connections, users are able to start, develop and deploy projects online, by taking the benefit of the processing power of their servers. Developers, Testers and Managers are able to follow the development of the project from wherever they are by using small notebook PCs or tablets. The aim is to provide users with full control on their servers and at the same time to make it possible to use the server resources in an efficient way.

This system will also be made available as an open source to people or institutions that might find a good use of it and can also make modifications or add new features based on their personal preferences. The maintenance of the project will be supported by donations.

On later stages I am planning to develop and integrate an Application Center where users can directly get and install apps to their server from here and developers can contribute by creating necessary apps or plugins based on their usage needs. Apps or plugins will be evaluated based on some criteria.

In order to promote the system, I am planning to offer some appropriate apps used in Web Server Management System as services on cloud, because our system was initially designed keeping this in mind. Apps like Code Editor together with other Software Project Management Tools will be offered on cloud for free in order to support the development of new apps for our system which are related to the server management needs on different countries and also to develop open source projects online individually or in a group.



## REFERENCES

- [1] Operating System,  
*[http://gama.vtu.itlibliotekalOperating\\_systems/Operating\\_systems.pdf](http://gama.vtu.itlibliotekalOperating_systems/Operating_systems.pdf)*, 2012
- [2] Features of OS, *[http://computersecuritycrew.com/features\\_oCos.php](http://computersecuritycrew.com/features_oCos.php)*, 2012
- [3] Dr. Kumar Saurabh, "*Cloud Computing-Insights into New-Era Infrastructure*",  
New Delhi, Wiley India Private Limited, 2011
- [4] Cloud computing tutorial, *<http://thecloudtutorial.com>*, 2012
- [5] NIST definition of cloud services, *<http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>*, 2012
- [6] What is web server,  
*[http://www.webdevelopersnotes.com/basics/what\\_is\\_web\\_server.php](http://www.webdevelopersnotes.com/basics/what_is_web_server.php)*, 2012
- [7] "*Distributed Application Architecture*". Sun Microsystem. 2009
- [8] Bhuvan Uргаonkar, "*Dynamic Resource Management In Internet Hosting Platforms*", 2005
- [9] "*Web Server Software Architectures*". IEEE Internet Computing, 2004, Paper
- [10] Edith Hofer, "*Component-based Web Development*", Graz University of  
Technology, Austria, 2007
- [11] jQuery User Interface "*<http://jqueryui.com>*"
- [12] elFinder – file manager for web "*<http://elfinder.org>*"
- [13] phpMyAdmin – handles the administration of MySQL for web  
*<http://www.phpmyadmin.net>*"
- [14] CodeMirror – In-browser code editing "*<http://codemirror.net>*"
- [15] Jwysiwyg – Web based text editor "*<https://github.com/akzhan/jwysiwyg>*"

[16] Shellinabox – Web based Ajax terminal “<http://code.google.com/p/shellinabox>”

[17] FullCalendar – Web based Ajax calendar “

<http://arshaw.com/fullcalendar>”

[18] Flot – Graph plotting for jQuery “<http://code.google.com/p/flot>”

[19] jQuery Easing Plugin – “<http://gsgd.co.uk/sandbox/jquery/easing>”

APPENDIX A

Control panel	Licence	Free	Open Source	Last Release	Linux	Windows	Frontend	Backend	Plugin Support	IPV6 Support	Multiserver Support
Baifox	GPL	Yes	Yes	4/19/2009	Yes	No	PHP	PHP, SQLite	?	?	?
cPanel	Proprietary	No	No	9/10/2011	Yes	Partial	Perl	Perl	Yes	?	?
DirectAdmin	Proprietary	No	No	9/1/2011	Yes	No	C++	C++, scripts, PHP	Yes	Partial	?
Domain Technologie Control	GNU LGPL	Yes	Yes	9/1/2010	Yes	No	PHP	?	?	No	?
Gnupanel	GPL	Yes	Yes	3/21/2012	Yes	No	PHP	PHP, PgSQL	?	?	?
H-Sphere	Proprietary	No	No	5/25/2011	Yes	Yes	Java	Java	Yes	?	Yes
HDE Controller X	Proprietary	No	No	11/1/2011	Yes	No	PHP	PHP	?	Yes	Yes
Hosting Controller	Proprietary	No	No	9/5/2011	Yes	Yes	.NET	MSSQL	Yes	No	?
i-MSCP	GPL	Yes	Yes	10/25/2011	Yes	No	PHP	Perl	Yes	Yes	No
InterWorx	Proprietary	No	No	11/7/2011	Yes	No	PHP	PHP, MySQL	Yes	No	Yes
ISPConfig	BSD	Yes	Yes	2/13/2012	Yes	No	PHP	PHP, MySQL	Yes	Yes	Yes
ispCP	GPL	Yes	Yes	11/24/2010	Yes	No	PHP	Perl	No	No	No
Kloxo / Lxadmin	AGPL	Yes	Yes	11/13/2011	Yes	No	PHP	PHP, MySQL	No	No	Yes
Plesk	Proprietary	No	No	1/12/2012	Yes	Yes	PHP	PHP/MySQL	?	Yes	?
SysCP	GPL	Yes	Yes	5/11/2010	Yes	No	PHP	PHP, MySQL	Yes	Yes	No
Froxlор	GPL	Yes	Yes	1/18/2012	Yes	No	PHP	PHP, MySQL	Yes	No	Partial
Usermin	BSD style	Yes	Yes	10/2/2011	Yes	No	?	?	?	?	?
Virtualmin	GPL	Yes	Yes	10/2/2011	Yes	No	Perl	Perl	Yes	Yes	Partial
Virtualmin Pro	Proprietary	No	No	10/2/2011	Yes	No	Perl	Perl	Yes	Yes	Partial
Webmin	BSD style	Yes	Yes	1/21/2012	Yes	Partial	Perl	Perl	Yes	Yes	Yes

Information on existing Control Panels for managing web servers

## VITA

Name: Elton Demirxhiu

Email Address: edemirxhiu08@epoka.edu.al

Education: Computer Engineering, Epoka University