**LAMP STACK-WEB SERVER CONFIGURATION**

LAMP stands for Linux, Apache, MySQL and PHP. Lamp Stack is a popular web development platform commonly used to run dynamic websites. It was one of the first open source software stacks for the web and remains one of the most common ways to deliver web applications. Each component in LAMP contributes essential capabilities to the stack.

**LAMP:** The LAMP components are largely interchangeable and not limited to the original selection. As a solution stack, LAMP is suitable for building [dynamic web sites](https://en.wikipedia.org/wiki/Dynamic_web_site) and [web applications](https://en.wikipedia.org/wiki/Web_application).

Since its creation, the LAMP model has been adapted to other componentry, though typically consisting of [free and open-source software](https://en.wikipedia.org/wiki/Free_and_open-source_software). For example, an equivalent installation on the Windows family of operating systems is known as [WAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)#WAMP) and an equivalent installation on macOS is known as [MAMP](https://en.wikipedia.org/wiki/MAMP).

LAMP server is a type of software bundle found in a specific server. LAMP stands for Linux, Apache, MySQL, and PHP. The P can also stand for Perl or Python but generally it is assumed to be PHP. It is a solution stack of software programs, commonly open-source programs, used together to run dynamic websites or servers. Linux is the operating system (sort of like a windows system). Apache is the web server used. MySQL is the database management system. PHP, Perl, or Python is used as the programming language for the system. Though the originators of these open source programs did not design them all to work specifically with each other, the combination, has become popular because of its low acquisition cost and because of the ubiquity of its components (which come bundled with most current Linux distributions). When used in combination they represent a solution stack of technologies that support application servers.

**LINUX: THE OPERATING SYSTEM**

Linux is a free and open source operating system (OS) that has been around since the mid-1990s. Today, it has an extensive worldwide user base that extends across industries. Linux is popular in part because it offers more flexibility and configuration options than some other operating systems. Largely known for its use in servers, it is used as an operating system for a variety of computer hardware, such as desktop computers, supercomputers, and embedded devices such as mobile phones and routers.

**APACHE: THE WEB SERVER**

Robert McCool created the first Apache web server. He was involved with the National Center for Supercomputing Applications web server, also known as NCSA HTTPD. Robert McCool left the HTTPD in the mid. Development of the server technology stalled as a result of his absence. Several other people assisted McCool with his development of the server. The Apache web server processes requests and serves up web assets via HTTP so that the application is accessible to anyone in the public domain over a simple web URL. This apache server runs a large share of websites currently on internet.

**MYSQL: THE DATABASE**

It is an open source relational database management system for storing application data. With My SQL, you can store all your information in a format that is easily queried with the SQL language. MySQL is suitable for running even large and complex sites. Libraries for accessing MySQL databases are available in all major programming languages with language-specific API’s. In addition, an ODBC interface called MYODBC allows additional programming languages that support the ODBC interface to communicate with a MySQL database, such as ASP or Cold fusion. The MySQL server and official libraries are mostly implemented in ANSI C.

**PHP: THE SCRIPTING PROGRAMMING LANGUAGE**

The PHP open source scripting language works with Apache to help you create dynamic web pages. You cannot use HTML to perform dynamic processes such as getting the data from the database. To provide this type of functionality, you simply drop PHP code into the parts of a page that you want to be dynamic.

LAMP has a layered architecture, with Linux at the lowest level. The next layer is Apache and MySQL, followed by PHP. Although PHP is nominally at the top or presentation layer, the PHP component is inside Apache.

**HOW LAMP WORKS:**

The process starts when the Apache web server receives requests for web pages from a user’s browser. If the request is for a PHP file, Apache passes the request to PHP, which loads the file and executes the code contained in the file. PHP also communicates with MySQL to fetch any data referenced in the code. PHP then uses the code in the file and the data from the database to create the HTML that browsers require to display web pages.

The LAMP stack is efficient at handling not only static web pages, but also dynamic pages where the content may change each time it is loaded depending on the date, time, user identity and other factors.

After running the file code, PHP then passes the resulting data back to the Apache web server to send to the browser. It can also store this new data in MySQL. And of course, all of these operations are enabled by the Linux operating system running at the lowest level of the stack.

**INSTALLATION LAMP ON LINUX(CENTOS7):**

**Installation of apache web server:**

The first step of assembling the LAMP stack is to install the web server. We can install webserver using yum package manager.

command: sudo yum install httpd

Start apache server by running the following command.

command: sudo systemctl start httpd. Service

httpd is the Apache Hyper Text Transfer protocol server program. It is designed to run as a standalone daemon process. When used like this it will create a pool of child processes or threads to handle requests.

Check whether the service is running by going to your server’s public IP address. The browser should display the test CentOS 7 Apache web page.

**Installation of MYSQL(MARIADB):**

After Apache webserver installation we do MySQL installation. The installation process is as follows.

Command**:** sudo yum install mariadb-server mariadb

When a y/n prompt appears, confirm with y.

Now start MariaDB using the command:

sudo systemctl start mariadb

**Running MySQL security script:**

MariaDB does not have secure settings by default. Therefore, you need to configure settings, test the database, and remove anonymous users.

We do this by typing the command:

sudo mysql\_secure\_installation

It will be asked to provide Mariadb root password and some set of queries, so provide them.

Lastly, enable MariaDB to start up when you boot the system:

sudo systemctl enable mariadb.service

**Installation of PHP:**

Install the MySQL extension along with PHP, using the yum package manager with the command:

sudo yum install php php-mysql

To have your Apache webserver start working along with PHP, restart the server:

sudo systemctl restart httpd. service

we test php processing by writing simple php code in /var/www/html/filename. php and by visiting http://ip\_address/filename.php

If a firewall is enabled you will need to open a route for HTTP traffic.

**Restart apache**:

sudo systemctl restart httpd

This is about installing and configuring LAMP STACK.

**Accessing the MySQL database:**

We can access the MySQL database by using the following command.

command: mysql -u root -p

Then it will ask for password, by giving it we can access databases. We can access the database with the browser.

With the phpMyAdmin (before we need to install phpMyAdmin on Lamp) n we can access the database through browser by username and password.

**PHP Files:**

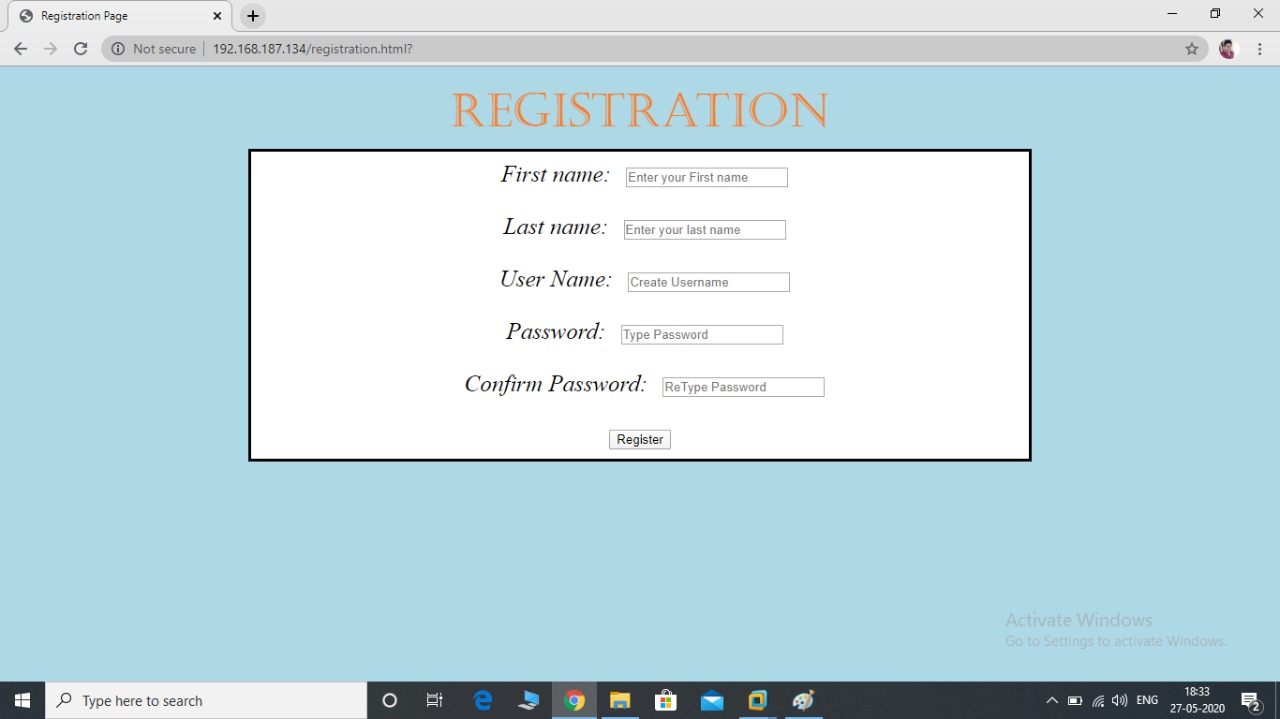
All the files which are needed for website either it is static or dynamic, there should be in /var/www/html. It can contain sub directories and php files for website development.

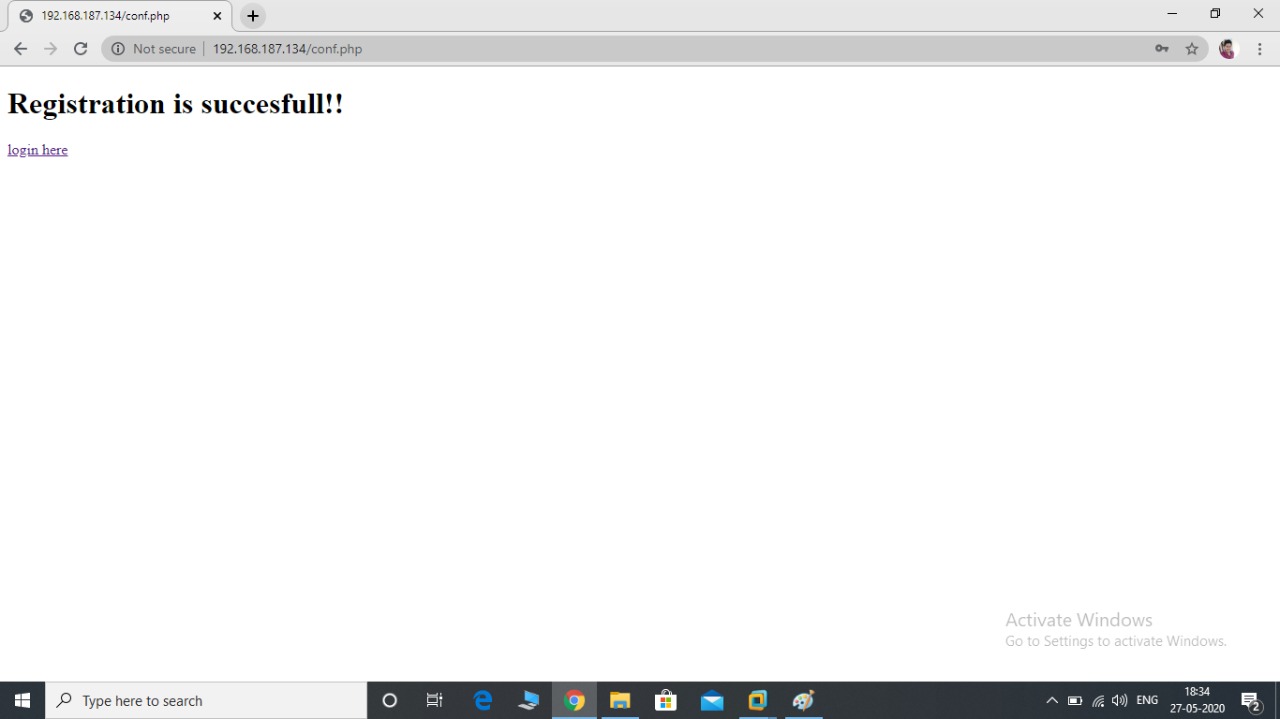
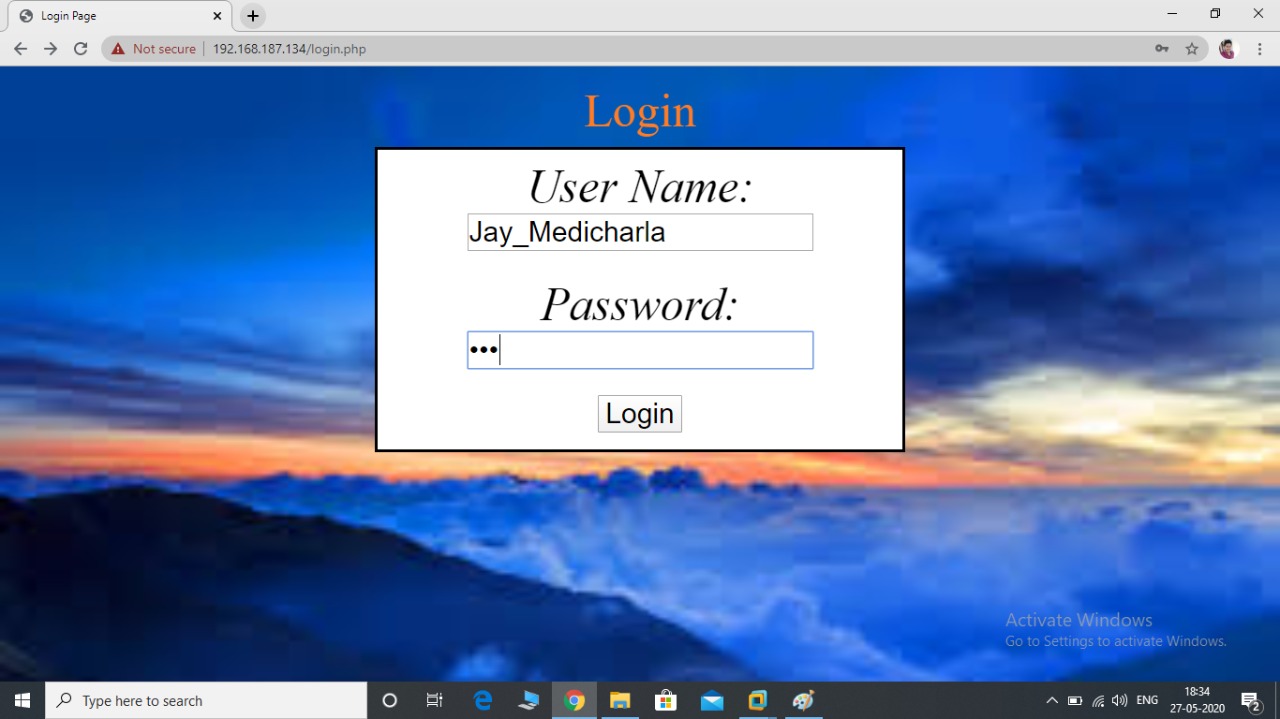
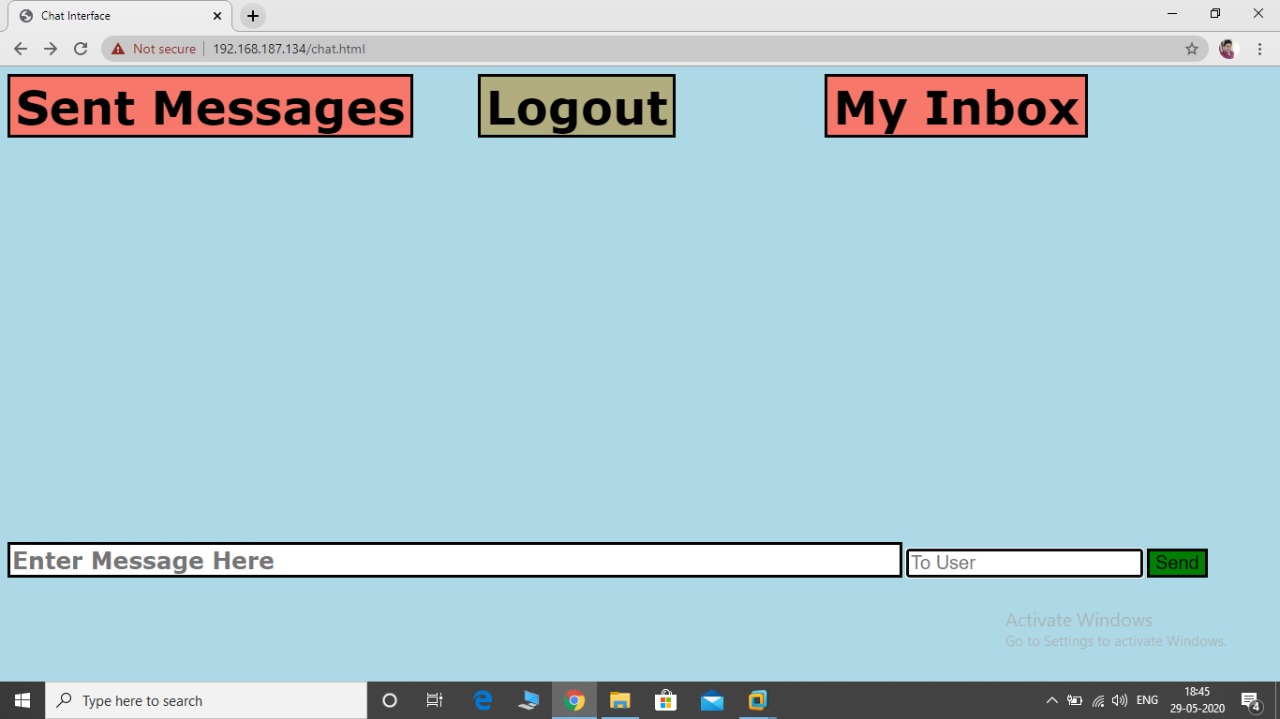
**FEATURES OF OUR MAIN INTERFACE:**

1. First comes a highly secured welcome website, where we need to get register. This website is designed with HTML and CSS.



1. By clicking on Register button, we will be having registration page. Where we need to enter all the details such as First name,

Last name, User name, Password along with Confirm Password.

1. After get registered we will be notifying that “Registration is successful”.
2. If the person is already got registered, he/she can go with login by clicking “Login here” button.
3. After login, we will be having “Chat Interface” where one user can communicate with another user.
4. In this Interface, we are having 3 system protocols.

* Sent Messages
* Logout
* My Inbox

1. Sent Messages: After clicking this button, we can be able to see the messages which were sent by us.
2. My Inbox: In this, we can able to see the messages which were sent by others.
3. Logout: Through this button we can exit from the interface.

**CONCLUSION:**

We have been configured the LAMP Stack with Linux Operating System to develop such a dynamic website which helps the user to communicate with others. This really helps for the users to communicate and also their data will be safe under this website. No cyber hacks can be allowed in this. And this interface can able to run on any different platforms as here we are using Apache as a web server.