**Technical Reference Manual**

Technologies needed:

1. Openfire server version 3.10.2
2. XAMPP or WAMP server having Apache and MySQL modules
3. Web browser: Chrome or Firefox
4. .war file for the web application
5. .jar file for the audio desktop application
6. **Openfire server version 3.10.2**

**Openfire installation and deployment on system**

Running Openfire in Windows:

If you used the Openfire installer, a shortcut for starting the graphical launcher is provided in your Start Menu. Otherwise, run openfire.exe in the **bin/** directory of your Openfire installation. A button on the on the launcher allows you to automatically open your web browser to the correct URL to finish setting up the server:

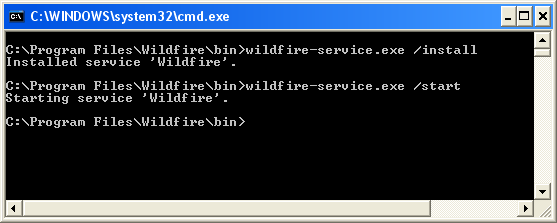


**Windows Service**

If you're running Openfire on Windows, you will likely want to run Openfire as a standard Windows service after initial setup. If you used the Windows installer, a **openfire-service.exe** file will be in the **bin** directory of the installation. You can use this executable to install and control the Openfire service.

From a console window, you can run the following commands:

* **openfire-service /install** -- installs the service.
* **openfire-service /uninstall** -- uninstalls the service.
* **openfire-service /start** -- starts the service
* **openfire-service /stop** -- stops the service.



You can also use the Services tool in the Windows Control Panel to start and stop the service.

We have the following 5 steps to complete the configuration:

1. Language Selection

2. Server Settings

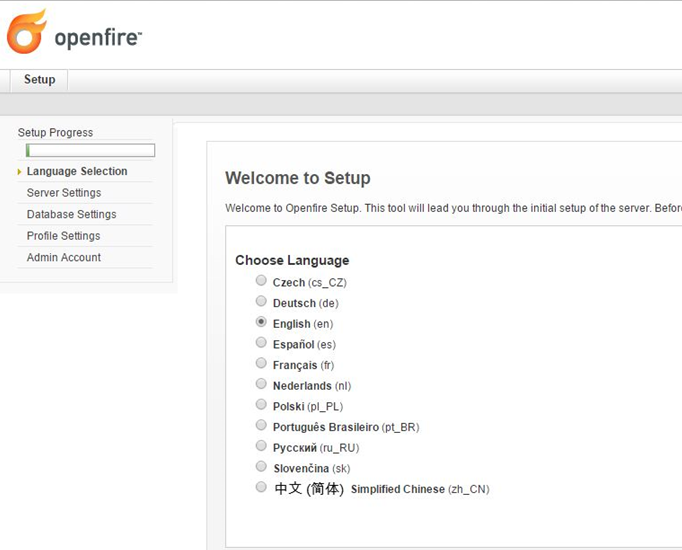
3. Database Settings

4. Profile Settings

5. Admin Accounts

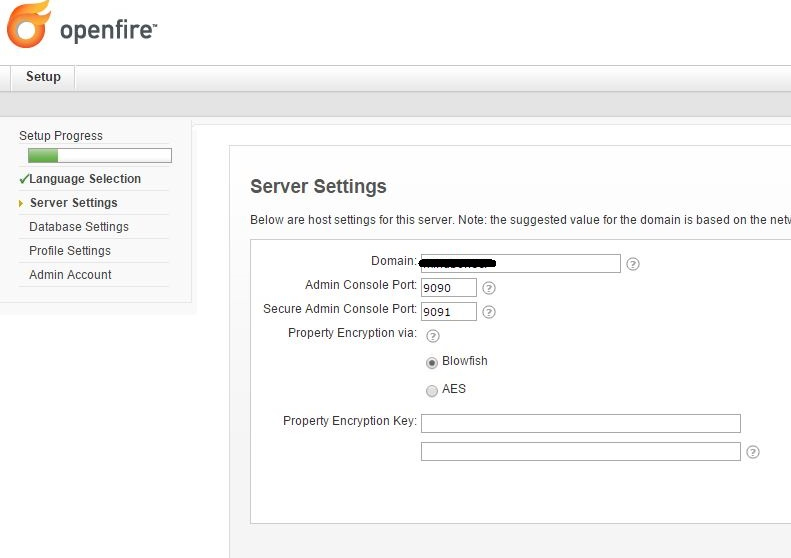
**Step 1: Language Selection**

Choose the language which you want the installer and the admin console to be in and then click on ‘Continue’.



**Step 2: Server Settings**

In this step we configure the server domain, admin console port, and also the secure admin port. Generally, we do not change these data, unless and until we need custom port. But make sure that these ports are open on your network.



**Step 3: Database Settings**

Here we have 2 ways to setup the database:

Embedded Database

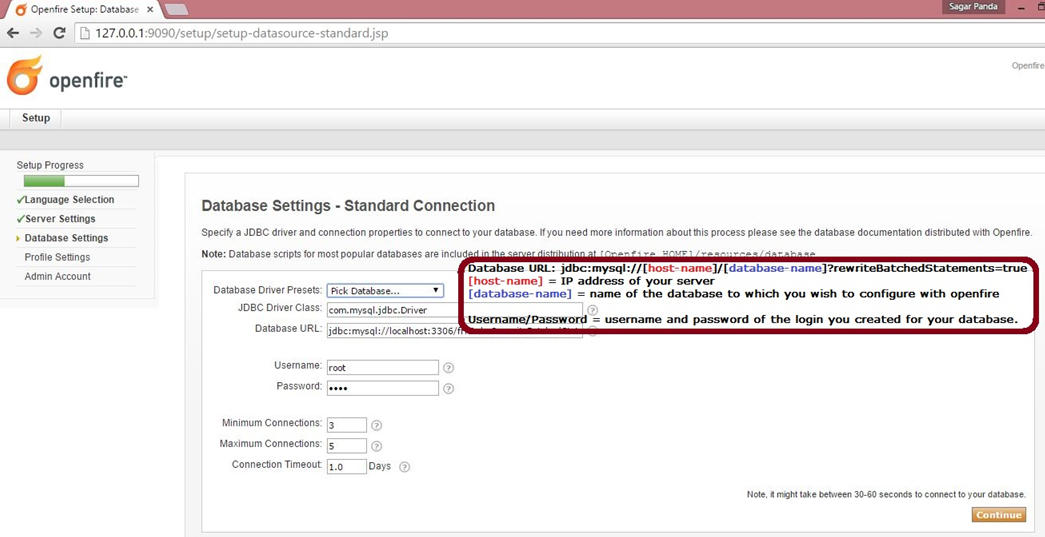
Standard Database Connection

**Embedded Database:** Embedded database is the easiest way to configure and setup as it does not require any external database. However, it does not give the same level of performance as given by an external database.

If you choose to use Embedded Database and click continue then it will take you to Profile settings.

**Standard Database:** Standard Database connection requires an external database up and running.

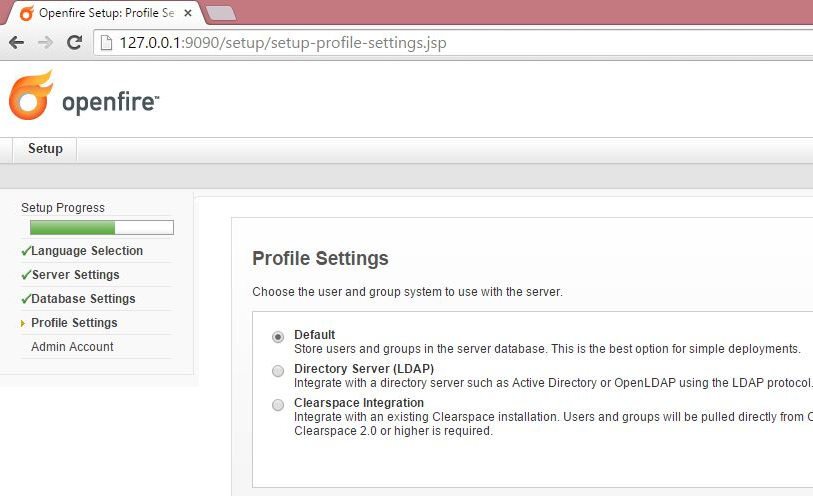
By default Standard Database is selected, click on continue to configure to external database.



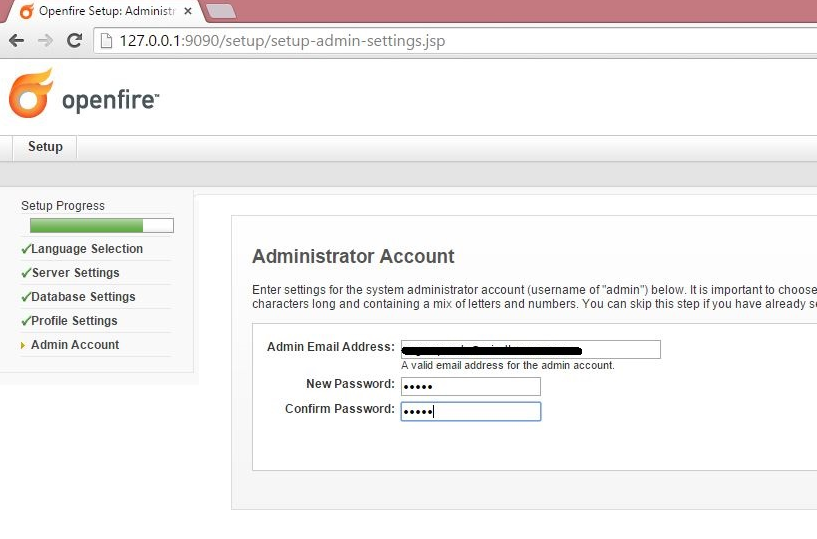
**Step 4: Profile Settings**

It allows you to choose methods for storing and authenticating users and group data.

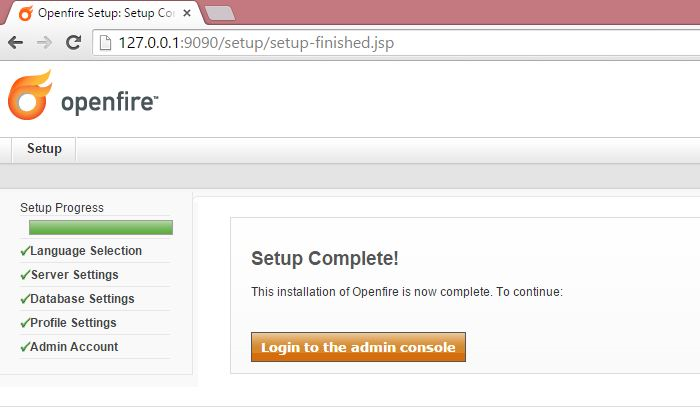
For more simplicity, I rely on default option and continue.



**Step 5: Admin Account**



This is the final step of our web base configuration. Make sure you enter a valid email address for your admin. The default password for admin is ‘admin’. Enter your new password and confirm password and continue to proceed to see the Setup Complete message.



Now, if we try to login to the admin console, it will give the following error: **“Login failed: make sure your username and password are correct and that you are an admin or moderator”** Even though your credentials are correct, it won’t allow us to login.

Don’t worry, we have a solution for it.

Go to the openfire server database,(this is the external standard database which we configured in Step 3 )

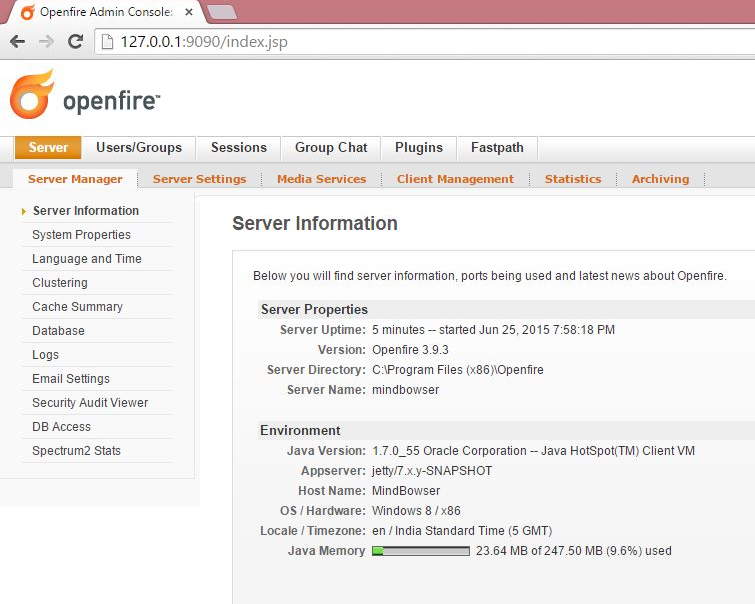
Check the ofUser table which contains the list of users that we register/add on our openfire.

Here we need to change the password of admin by simply firing the below query:



You can set plain Password of your own choice.

Now you can login to the admin console (http://127.0.0.1:9090/) using the username as admin and password as this new plain password.



Once the setup is completely done, you can check your database to which we have linked openfire. You can see that new tables have been added to your database by the openfire. For a complete list of database tables and schema you can read [here](http://www.igniterealtime.org/builds/openfire/docs/3.10.0/documentation/database-guide.html).

**Openfire Admin Console:**

You have an entire control on your openfire server through the admin console. Let’s look at some of important and most used properties and functionalities which can be achieved through admin console.

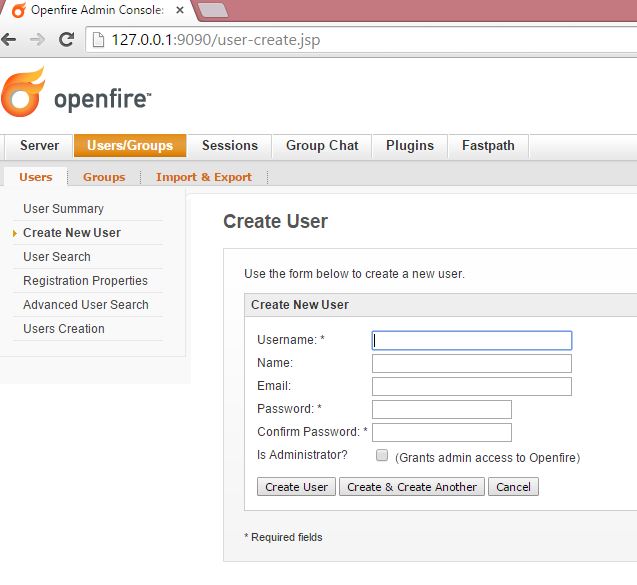
**Creating a user:**

In the admin console:

1. Click on **User/Groups**

2. Under Users section click on **Create New User**

3. Fill all the required details and click **Create User**

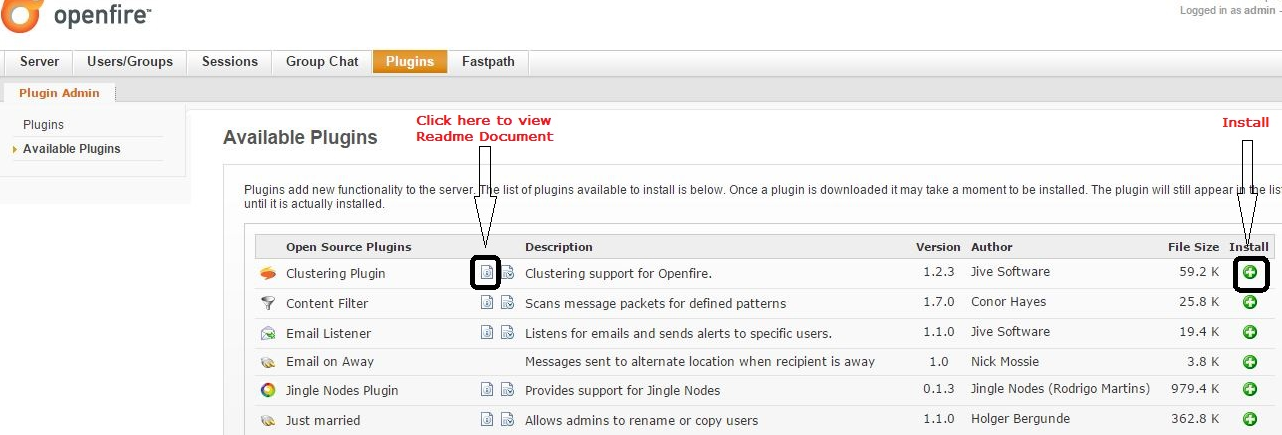


Then, under **User Summary** you can see the newly created user.

This was one way of creating user through the admin console. Openfire has provided us with the functionality of REST api’s for various operations like creating/deleting/modifying user, adding/deleting a user to/from a roster of another user and many other operations can be performed easily by the hit of a url with some parameters added to it.

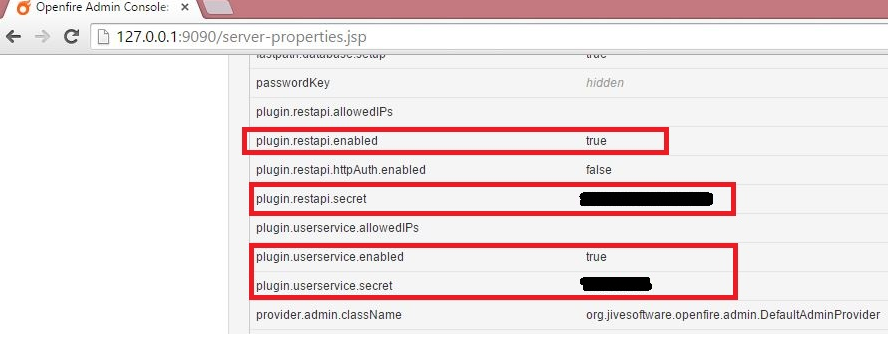
For this you will require functionality specific **PLUGINS** to be installed in the openfire server.

Every plugin has its own functionality which has been mentioned clearly in its readme document.



You can install plugins by simply clicking on the install button. Once installed, restart the openfire server. You can also see jar files of the installed plugins in the installation directory of openfire server, generally which is C:\Program Files (x86)\Openfire\plugins.

While using REST or User Service we will require the secret keys for security purpose, which you can find it under Server ⇒ Server Manager ⇒ System Properties.



System properties are an aggregate of all the properties and settings of the openfire. Meanwhile you can also check and edit any specific properties under Server ⇒ Server Settings.

1. **XAMPP or WAMP server having Apache and MySQL modules**

XAMPP stands for Cross-Platform (X), Apache (A), MySQL (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing purposes. Everything you need to set up a web server – server application (Apache), database (MySQL), and scripting language (PHP) – is included in a simple extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server is extremely easy as well.

**Installing XAMPP**

Follow these steps for installing XAMPP:

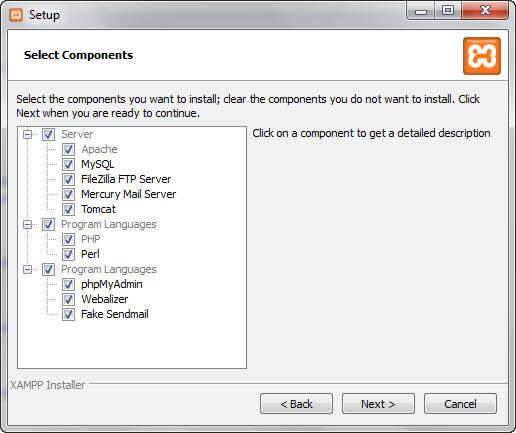
**Step 1:** Disable your anti-virus as it can cause some XAMPP components to behave erratically.

**Step 2**: Disable User Account Control (UAC). UAC limits write permissions to XAMPP’s default installation directory (c:/Program Files/xampp), forcing you to install in a separate directory. You can learn how to disable UAC here. (Optional)

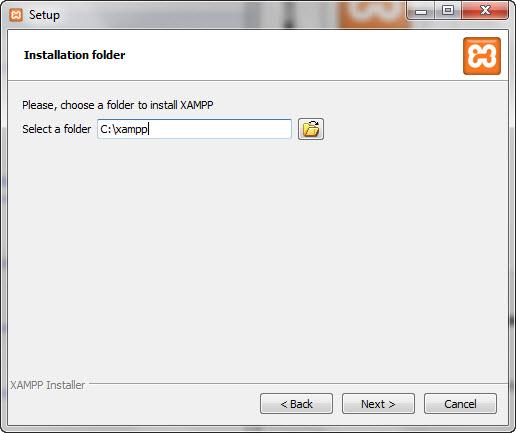
**Step 3:** Start the installation process by double-clicking on the XAMPP installer. Click ‘Next’ after the splash screen.



**Step 4**: Here, you can select the components you want to install. Choose the default selection and click ‘Next’.



**Step 5**: Choose the folder you want to install XAMPP in. This folder will hold all your web application files, so make sure to select a drive that has plenty of space.

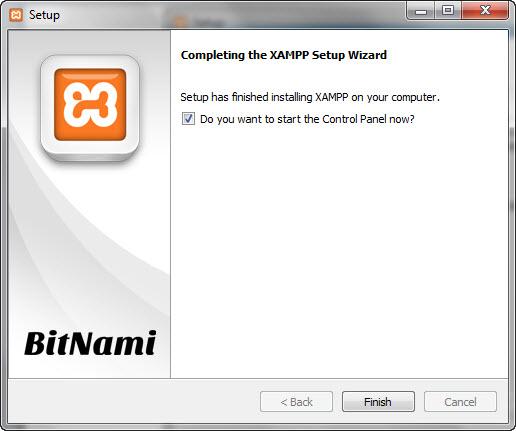


**Step 6:** The next screen is a promo for BitNami, an app store for server software. Deselect the ‘Learn more about BitNami for XAMPP’ checkbox, unless you actually enjoy receiving promo mails!



**Step 7:** Setup is now ready to install XAMPP. Click Next and wait for the installer to unpack and install selected components. This may take a few minutes. You may be asked to approve Firewall access to certain components (such as Apache) during the installation process.

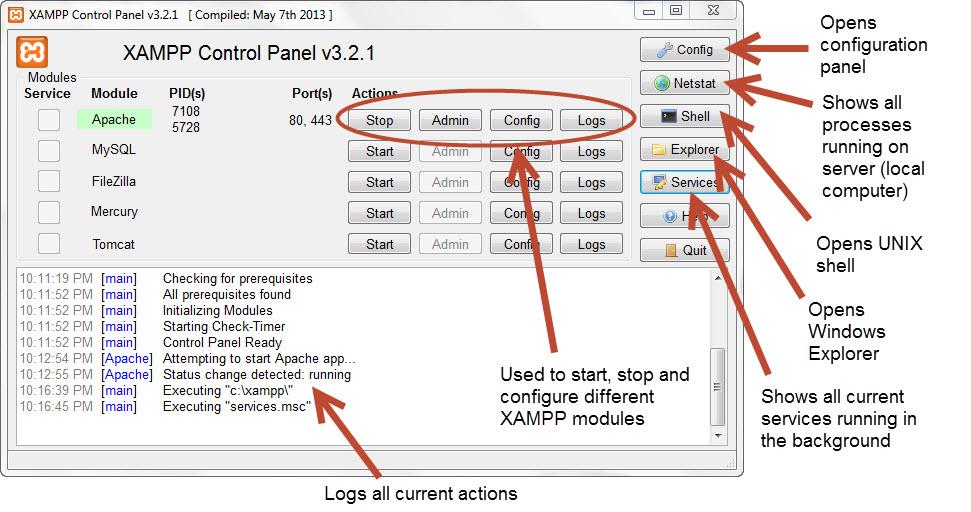
**Step 8:** Installation is now complete! Select the ‘Do you want to start the Control Panel now?’ checkbox to open the XAMPP control panel.



**Understanding XAMPP Control Panel**

The XAMPP control panel gives you complete control over all installed XAMPP components. You can use the CP to start/stop different modules, launch the Unix shell, open Windows explorer and see all operations running in the background.

Here is a quick overview of the Control Panel. For now, you only need to know how to start and stop an Apache server.



**Testing Your XAMPP Installation**

Follow these steps to test your XAMPP installation by launching the Apache web server and creating a simple PHP file.

**Step 1:** In the XAMPP control panel, click on ‘Start’ under ‘Actions’ for the Apache module. This instructs XAMPP to start the Apache webserver.

**Step 2:** Open your web browser and type in: [**http://localhost**](http://localhost/) or 127.0.0.1

**Step 3:** Select your language from the splash screen.



**Step 4:** You should see the following screen. This means you’ve successfully installed XAMPP on your computer.



Thus, we have successfully set up a Xampp local Server on the local machine.

1. **Web browser: Chrome or Firefox**

As OpenFire Server is best compatible with Google Chrome and the default browser in the configuration files of OpenFire is Google Chrome, so it is advisable to use Google Chrome. To use Firefox, additional changes will be required to be done in the configuration file.

1. **.war file for the web application**

A war (web archive) File contains files of a web project. It may have servlet, xml, jsp, image, html, css, js etc. files.

**Creating war file:**

To create war file, you need to use jar tool of JDK. You need to use -c switch of jar, to create the war file.

Go inside the project directory of your project (outside the WEB-INF), then write the following command:

jar -cvf projectname.war \*

Here, -c is used to create file, -v to generate the verbose output and -f to specify the arhive file name.

The \* (asterisk) symbol signifies that all the files of this directory (including sub directory).

**Deploying war file:**

There are two ways to deploy the war file.

1. By server console panel
2. By manually having the war file in specific folder of server.

If you want to deploy the war file in apache tomcat server manually, go to the webapps directory of apache tomcat and paste the war file here.

Now, you are able to access the web project through browser.

1. **.jar file for the audio desktop application**

In Java, it is common to combine several classes in one .jar ("java archive") file.  Library classes are stored that way.  Larger projects use jar files.  You can create your own jar file combining several classes, too. jar files are created using the jar.exe utility program from the JDK.  You can make your jar file runnable by telling jar.exe which class has main.  To do that, you need to create a manifest file.  A manifest is a one-line text file with a "Main-Class" directive.

**Creating a JAR File**

The basic format of the command for creating a JAR file is:

jar cf *jar-file input-file(s)*

The options and arguments used in this command are:

* The c option indicates that you want to *create* a JAR file.
* The f option indicates that you want the output to go to a *file* rather than to stdout.
* jar-file is the name that you want the resulting JAR file to have. You can use any filename for a JAR file. By convention, JAR filenames are given a .jar extension, though this is not required.
* The input-file(s) argument is a space-separated list of one or more files that you want to include in your JAR file. The input-file(s) argument can contain the wildcard \* symbol. If any of the "input-files" are directories, the contents of those directories are added to the JAR archive recursively.

The c and f options can appear in either order, but there must not be any space between them.

This command will generate a compressed JAR file and place it in the current directory. The command will also generate a [default manifest file](https://docs.oracle.com/javase/tutorial/deployment/jar/defman.html) for the JAR archive.