Start a New Zend Framework 2 Project

XAMPP makes it easy to start developing with PHP, and Zend Framework 2 is one of the most popular PHP development frameworks. This guide walks you through the process of initializing a new Zend Framework 2 project with XAMPP.

NOTE This guide uses the command-line git client for Linux. If you don't already have this, you can install it easily by running the command *sudo apt-get install git* or *sudo yum install git* from your Linux terminal. It also assumes that the new Zend Framework 2 application will be accessible at the URL http://localhost/myapp/.

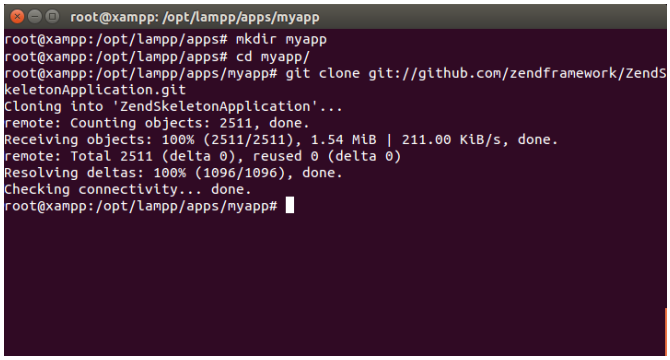
Follow these steps:

1. Open a new Linux terminal and ensure you are logged in as root.
2. Within your XAMPP installation directory (usually */opt/lampp*), create a new directory named *apps/* (if it doesn't already exist). Then, within this new *apps/* directory, create a directory to hold your Zend Framework 2 application and its related XAMPP configuration files. In this case, call the directory *myapp/*.

   ```
   cd /opt/lampp
   mkdir apps
   mkdir apps/myapp
   ```

3. Clone the Zend Framework 2 sample application repository to the *myapp/* directory using *git*.

   ```
   cd /opt/lampp/apps/myapp
   git clone git://github.com/zendframework/ZendSkeletonApplication.git
   ```

   This will produce a *ZendSkeletonApplication/* subdirectory in the *myapp/* directory. Rename this newly-created subdirectory to *htdocs*.

   ```
   cd /opt/lampp/apps/myapp/
   mv ZendSkeletonApplication htdocs
   ```

   NOTE This will be the main working directory for your Zend Framework 2 project.

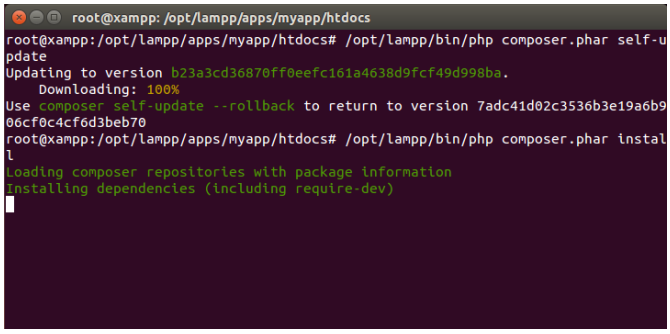
4. Change to the *myapp/htdocs/* directory and run the following commands to update Composer (the PHP dependency manager) and install the Zend Framework 2 components.

```
cd /opt/lampp/apps/myapp/htdocs
/opt/lampp/bin/php composer.phar self-update
/opt/lampp/bin/php composer.phar install
```

In case you encounter SSL errors when running the above commands, use the following command to let PHP know where to find your system's SSL certificates, then try again.

```
export SSL_CERT_DIR=/etc/ssl/certs
```

Here's an example of what you might see as Composer downloads and installs dependencies.



5. Next, within the *myapp/* directory, create a new conf/ subdirectory.

```
cd /opt/lampp/apps/myapp
mkdir conf
```

   a. Within the new *conf/* subdirectory, use your text editor to create and populate a file named *httpd-prefix.conf* with the following content:

```
Alias /myapp/ "/opt/lampp/apps/myapp/htdocs/public/"
Alias /myapp "/opt/lampp/apps/myapp/htdocs/public"
Include "/opt/lampp/apps/myapp/conf/httpd-app.conf"
```

   b. Within the *conf/* subdirectory, also create and populate a file named *httpd-app.conf* with the following content:

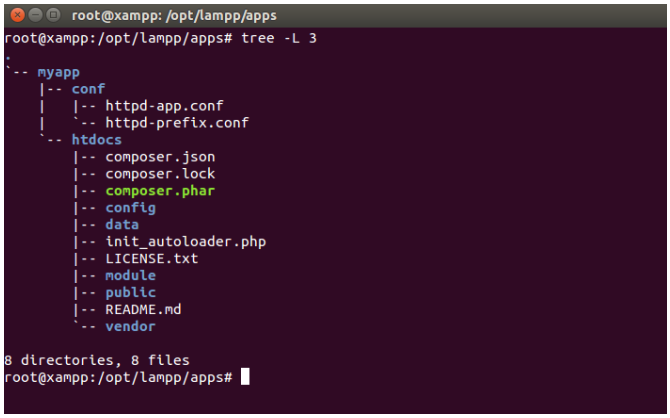
```
<Directory /opt/lampp/apps/myapp/htdocs/public>
  Options +FollowSymLinks
  AllowOverride All
  Require all granted
</Directory>
```

6. Edit the *httpd-xampp.conf* file in the *etc/extra/* subdirectory of your XAMPP installation directory and add the following line at the end to include the *httpd-prefix.conf* created earlier.

```
Include "/opt/lampp/apps/myapp/conf/httpd-prefix.conf"
```

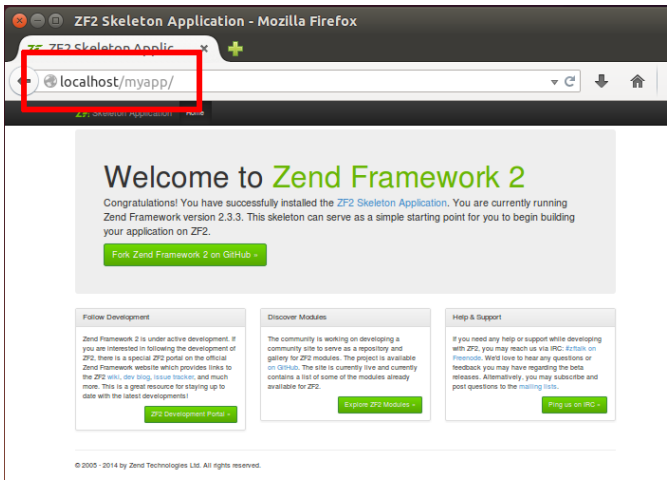
NOTE Remember to update the above file and directory paths so that they're valid for your system.

7. Check that you have a directory structure like this:



8. Restart the Apache server using the XAMPP control panel.

9. You should now be able to access the Zend Framework 2 skeleton application by browsing to http://localhost/myapp. Here is an example of the default welcome page you should see:



You can now begin developing your Zend Framework 2 application by modifying the skeleton application code. For more information, refer to the Zend Framework 2 User Guide.