Bitnami Wamp Installation

Created: 12/20/2018 Updated: 01/20/2022 Version: 2.0.1

Bitnami Stack Version: 8.0.13-0

Clean Your Machine

I start from a clean machine to ensure the stack and application install with the least number of hassles.

1. Uninstall Composer
2. Uninstall Previous stacks
3. Ensure port 81 is open for Apache and port 3306 is available for MySQL.
4. Ensure your path does not include a PHP interpreter.

Install Bitnami Stack

Bitnami has downloadable AMP stacks for multiple platforms. These stacks are under version control by Bitnami and the stack version is in the path name when the stack is installed on your machine. The version is also in the installation file name. There are different downloads for each type of platform, Windows, Mac & Linux, therefore the names of the download files are slightly different. Bitnami uses a prefix to specify the platform, WAMP, MAMP & LAMP. That prefix must be removed in the directory designation during install. The stack must be installed and is not directly downloaded from the project repository. Instead, we are relying on Bitnami to keep the versions synchronized. Go to: https://bitnami.com/stacks and search for WAMP then select the stack that is compatible with your development environment. Make sure the version is 8.0.13-0

1. Download Stack installation executable from Bitnami
2. Run the executable.
3. On the Select Components dialog, select only Laravel and phpMyAdmin.
4. On the Installation Folder dialog, select C:\Bitnami\wampstack-8.0.13-0 as the folder name.
5. On the Create MySQL 'root' Account dialog enter the word password as the password in two places; all lowercase.
6. On the Web Server Port dialog, select port 81
7. On Deploy Stack to Cloud… dialog, uncheck the box to deploy "cloud" and press next.
8. When ready to install, press next.
9. You will need to allow setup access through your firewall to set up Apache.
10. When setup is finished, launch the stack and pin the stack manager application to your taskbar. This is a very convenient app to manage the stack and is used frequently.
11. Verify you can open phpMyAdmin from the stack manager using a root account with password as the password.

12. Verify you can open the application from the stack manager and you get the default bitnami page at localhost:81
13. Select the Manage Servers tab in the stack manager and verify both servers are running and that you can stop and restart both servers.
14. If this all works, the stack is properly installed.

Run Artisan

1. Start a shell application
2. Change the directory to the Laravel folder
3. Verify that you can run php Artisan and get a list of commands.

Configure Apache

1. In the ../apache2/conf/bitnami/bitnami.conf file remove the comment designation to include: "C:/Bitnami/wampstack-8.0.13-0/frameworks/laravel/conf/httpd-prefix.conf"

2. In the ../apache2/conf/bitnami/bitnami.conf file change the document root and directory as follows (two places each):
   DocumentRoot "C:/Bitnami/wampstack-8.0.13-0/frameworks/laravel/public"
   <Directory "C:/Bitnami/wampstack-8.0.13-0/frameworks/laravel/public">

Install Composer

The Laravel Library and all the other open source libraries used in the Bitnami application are themselves under version control by the resective development groups. Therefore, once again, we must configure the platform manually to ensure synchronization between developers. Composer is the toll we use for that purpose. We install Composer after we have installed the stack because we want composer to use the PHP interpreter located in the stack.

1. Go to https://getcomposer.org/ and download install file
2. Run install
3. Do not select developer mode press next
4. Browse to: C:/Bitnami/wampstack-8.0.13-0/php/php.exe, select and press next
5. Do not use a proxy url
6. Composer will update your path to include the php interpreter
7. Click finish when done
8. Start a shell application
9. Change the directory to the Laravel folder
10. Verify that you can run composer and get a list of commands.

Install node.js and npm

Install node.js and npm; verify both of those apps run in the project directory using a shell app.

Configure Laravel project Bitnami Conf files

To ensure the project uses the correct index and view files make the following changes to files in the ../laravel/conf/bitnami directory:

1. In httpd-app.conf change “AllowOverride” to “All”
2. in httpd-vhosts.conf make sure “DocumentRoot” (two places) is set to: "C:\Bitnami\wampstack-8.0.13-0/frameworks/laravel/public"

Configure the Database

Make sure that PHPMyAdmin runs from localhost:81 and make the appropriate changes in the ../laravel/.env file to connect to the database.

Upgrade project

The open source libraries used by the project are under version control by their own development teams. Therefore, we will modify our composer.json file to include those libraries and ten run composer to upgrade the project to the specified levels.

1. Get …/laravel/composer.json
2. Start a shell application
3. Change the directory to the Laravel folder
4. Run composer upgrade
5. Verify project upgrades without error.

Extend project

At this point the Laravel project itself can be extended by including new requirements in the composer.json file like laravel/ui, laravelcollective/html, etc. New files such as Models, Views and Controllers can also be added manually to extend the capability of the project. Changes to other files like Kernel, Middleware, Providers, Routes, Database Migrations, etc. can also be made to change the behavior of the project as a whole.