

CP470: Lab #2 – Install & Configure Xdebug

Background:

Debugging tools are extremely useful in software development and comprise an important part of your development environment. Recall that debugging tools allow for code breakpoints and variable watches to be used. This allows us to step through our code line-by-line, and observe the values of variables in the process.

In class, we installed VS Code for Linux. If you haven't yet done so, follow the instructions here:

<https://code.visualstudio.com/Download>
<https://code.visualstudio.com/docs/setup/linux>

VS Code does not have built-in support for debugging PHP applications. Xdebug (xdebug.org) is an extension for the PHP language that enables debugging and other useful features. It uses a standard debugging protocol that is supported by VS Code. It will be installed as a PHP module on your LAMP server in this lab.

In the recommended setup, Xdebug has extensions for the Web browser and the code editor to communicate debugging information. There are official Xdebug extensions for Firefox and VS Code that will be installed and configured on your LAMP server in this lab.

Process:

Part 1: Install Xdebug & Configure PHP

(<https://xdebug.org/docs/install>)

- Check version of PHP and extensions:
 \$ php -v
- For Debian based systems, install Xdebug using apt:
 \$ sudo apt-get install php-xdebug
- Open Xdebug configuration file for editing:
 \$ sudo nano /etc/php/8.4/mods-available/xdebug.ini
- Add the following line to the files, if it isn't there already:
 zend_extension=xdebug.so
- Restart Apache:
 \$ sudo systemctl reload apache2
- Verify Xdebug extension is installed and configured:
 \$ php -v

Part 2: Activate Step Debugging

(https://xdebug.org/docs/step_debug)

- Open the configuration file from Step 1 and add the following lines:

```
xdebug.mode=debug
xdebug.start_with_request=yes
```
- Restart Apache:
\$ sudo systemctl reload apache2

Part 3: Install Extensions for Firefox and VS Code

(https://xdebug.org/docs/step_debug#browser-extensions)

- Xdebug officially supports numerous Web browsers and code editors
- Open the link below to in Firefox and install the extension (restart Firefox after installing):
(<https://addons.mozilla.org/en-GB/firefox/addon/xdebug-helper-for-firefox/>)
- Search for and install the “PHP Debug” by “Xdebug” extension in VS Code:
(<https://marketplace.visualstudio.com/items?itemName=xdebug.php-debug>)

Part 4: Test Debugging

- Open your website’s project folder in VS Code
- Open Firefox and browse to your website’s URL
- In VS Code, click on the Run and Debug tab (Ctrl+Shift+D) and select “create a launch.json file”
 - The launch.json file should have 3 configurations. Keep the one named “Listen for Xdebug”, and delete the others
- Next, place a breakpoint (indicated by a red dot) by clicking next to the line number in the PHP code file that you’d like to pause at
- Start the debugger from the Run and Debug tab by clicking on the green “play button”, or by pressing F5
- In Firefox, enable debugging on a page by clicking on the “bug” icon in the address bar, and selecting “Debug”
- Next, browse to the page that has the breakpoint. VS Code should pause execution of the PHP script when it is on the line with the breakpoint
- While the script is paused, you can inspect variable values, execute one statement at a time, add or remove breakpoints, and continue script execution

To Submit:

Submit to Blackboard **two screenshots**. One showing your browser with Developer Tools open to the Storage tab with Cookies selected. The other is of VS Code showing the debugger paused at a breakpoint in your message_get.php file. Examples are shown below:

message_get.php - my_domain - Visual Studio Code

File Edit Selection View Go Run Terminal Help

RUN AND DEBUG Listen for Xdebug ...

VARIABLES

Locals

- \$greeting = uninitialized
- \$i = uninitialized
- \$name = "Sean"
- \$num = 5

Superglobals

User defined constants

WATCH

CALL STACK Paused on step

{main} message_get.php 7:1

BREAKPOINTS

- Notices
- Warnings
- Errors

Accounts Options

- Everything
- message_get.php
- message_get.php

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Filter (e.g. text, |exclude, \escape)

Ln 7, Col 1 Spaces: 4 UTF-8 LF PHP

message_get.php

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "GET") {
    if (isset($_GET["name"]) && isset($_GET["num"])) {
        $name = $_GET["name"];
        $num = $_GET["num"];
        $greeting = "Hello $name. You look great today.";
    }
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PHP GET Message</title>
</head>
<body>
```

message_get.php - my_domain Terminal - sean@sean-Virt... 02:35 2025-02-05

