Purchase Service

Enable and configure Xdebug

Xdebug is an extension for PHP, and provides a range of features to improve the PHP development experience.

To enable it on the environment with PhpStorm and Docker, you need to set separate settings, for each project. **Before start, check that all needed containers are up and running.**

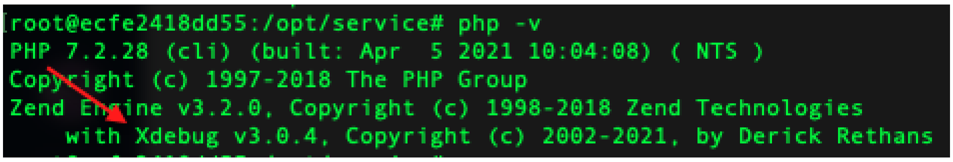
This manual applies to projects:
- Backoffice
- purchase-service
- settings-service
- auth-service

Prepare .env and enable Docker Xdebug

To start, copy Xdebug section from **.env.example** to **.env**
You need to set your APP_ENV=local, to get rid of PHP timeOut
Each project must have a different XDEBUG_REMOTE_PORT and XDEBUG_IDE_SERVER_NAME

```
XDEBUG_DEBUG=off
XDEBUG_PROFILE=off
XDEBUG_REMOTE_HOST=host.docker.internal
XDEBUG_REMOTE_PORT=9011
XDEBUG_IDE_SERVER_NAME=docker_purchase_server
```

Go to the project folder and run: **make xdebug-debug**
Open project bash: **make enter <project-name>**
Check that Xdebug is enabled: **php -v**



```
root@ecfe2418dd55:/opt/service# php -v
PHP 7.2.28 (cli) (built: Apr  5 2021 10:04:08) ( NTS )
Copyright (c) 1997-2018 The PHP Group
Zend Engine v3.2.0, Copyright (c) 1998-2018 Zend Technologies
    with Xdebug v3.0.4, Copyright (c) 2002-2021, by Derick Rethans
```

In case, when you need to disable Xdebug, run: **make xdebug-disable**

Configure PhpStorm

Add Docker Server

macOS

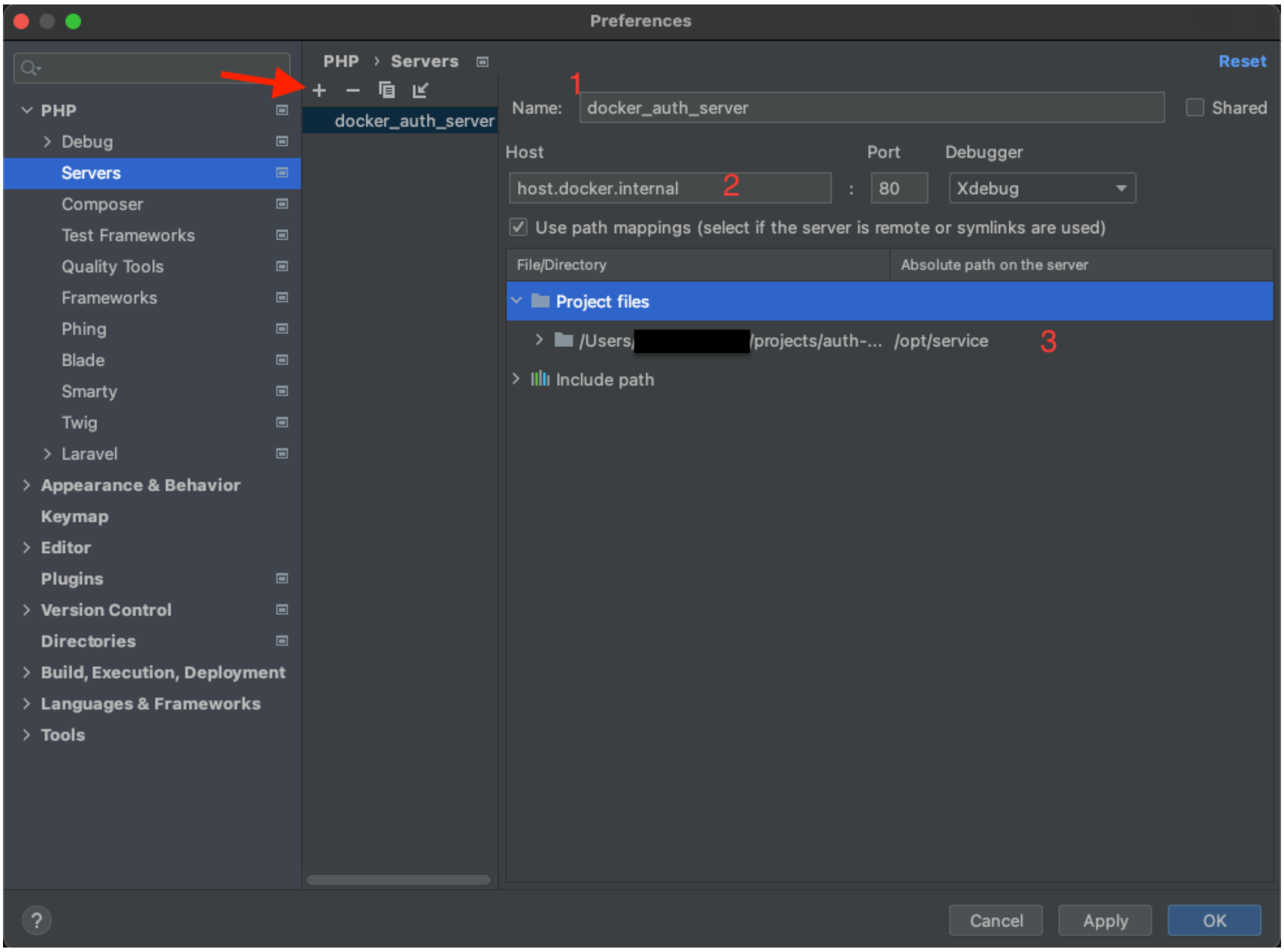
On the PhpStorm open ***Preferences -> PHP -> Servers*** click ***"+"***

1. Server Name from .env **XDEBUG_IDE_SERVER_NAME**, i.e. **docker_auth_server**
2. Host: **host.docker.internal**
3. Check "Use path mappings"
4. Absolute path on the server: **/opt/service**
   To find correct path, open docker bash:
   make enter <project-name>
   Find server path with: **pwd**

Linux

In many cases, with the latest Docker version, this will work for you, but some Linux and Windows user may have an issue here. For Linux, in some cases, Docker internal IP will be 172.17.0.1.

1. **You may need to add to /etc/hosts**
   172.17.0.1 host.docker.internal
2. **You can try to set**
   XDEBUG_REMOTE_HOST=172.17.0.1



Name: docker_auth_server □ Shared

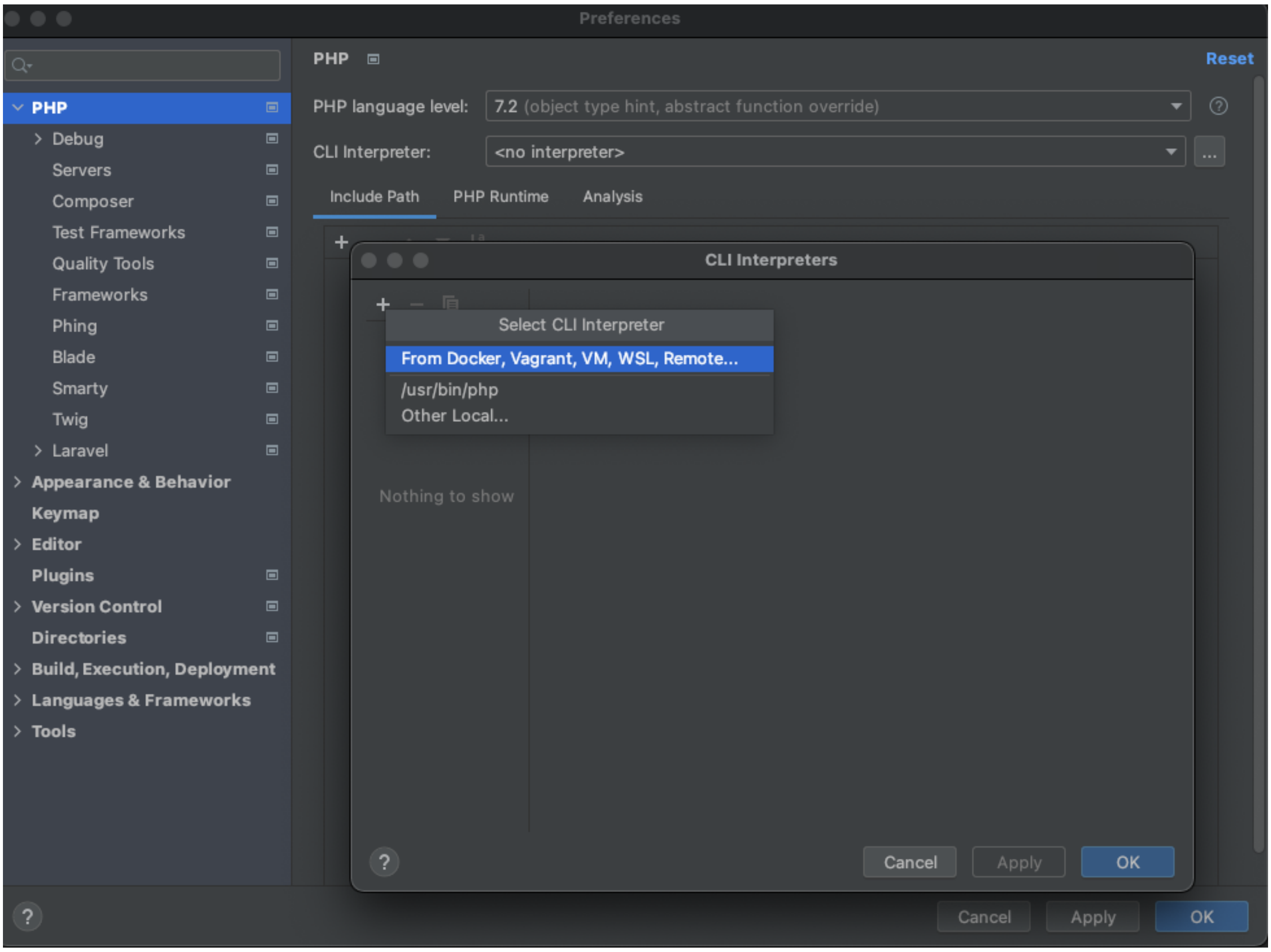
Host: host.docker.internal : Port: 80 Debugger: Xdebug

☑ Use path mappings (select if the server is remote or symlinks are used)

File/Directory	Absolute path on the server
Project files	
/Users/[redacted]/projects/auth-...	/opt/service

Setup the PHP CLI Interpreters

Go to ***Preferences -> PHP*** and click "..." (three dots) on CLI Interpreter, on popup click "+" and choose "From Docker, Vagrant, VM, WSL, Remote"



PHP language level: 7.2 (object type hint, abstract function override)

CLI Interpreter: <no interpreter>

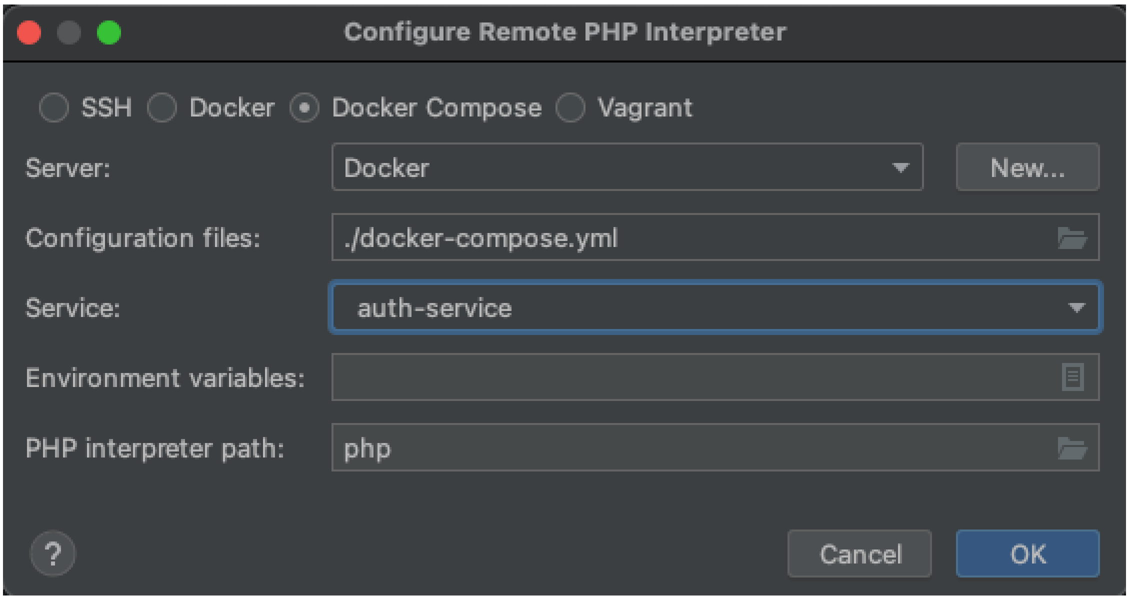
On the "*Configure Remote PHP Interpreter"* popup, choose **Docker Compose**

Server added

Server: Docker **(You don't need to add new Docker server for each project)**

Configuration files: ./docker-compose.yml

PHP interpreter path: php



○ SSH ○ Docker ● Docker Compose ○ Vagrant

Server: Docker

Configuration files: ./docker-compose.yml

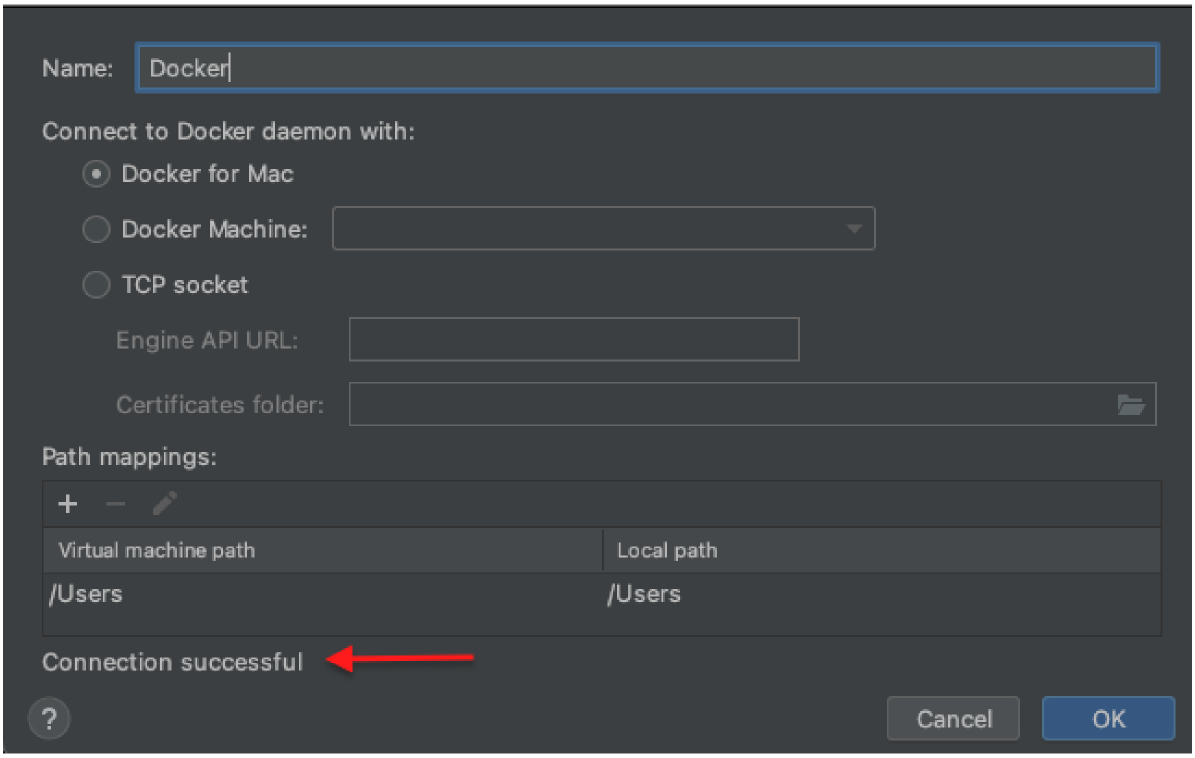
Service: auth-service

Environment variables:

PHP interpreter path: php

Add new Docker server

To create a new Docker connection, click "New..."
On most cases, Choose "Docker on {Mac, Linux, Windows}"
Wait until you got a message "Connection successful"



Name: Docker

Connect to Docker daemon with:

● Docker for Mac

○ Docker Machine:

○ TCP socket

Engine API URL:

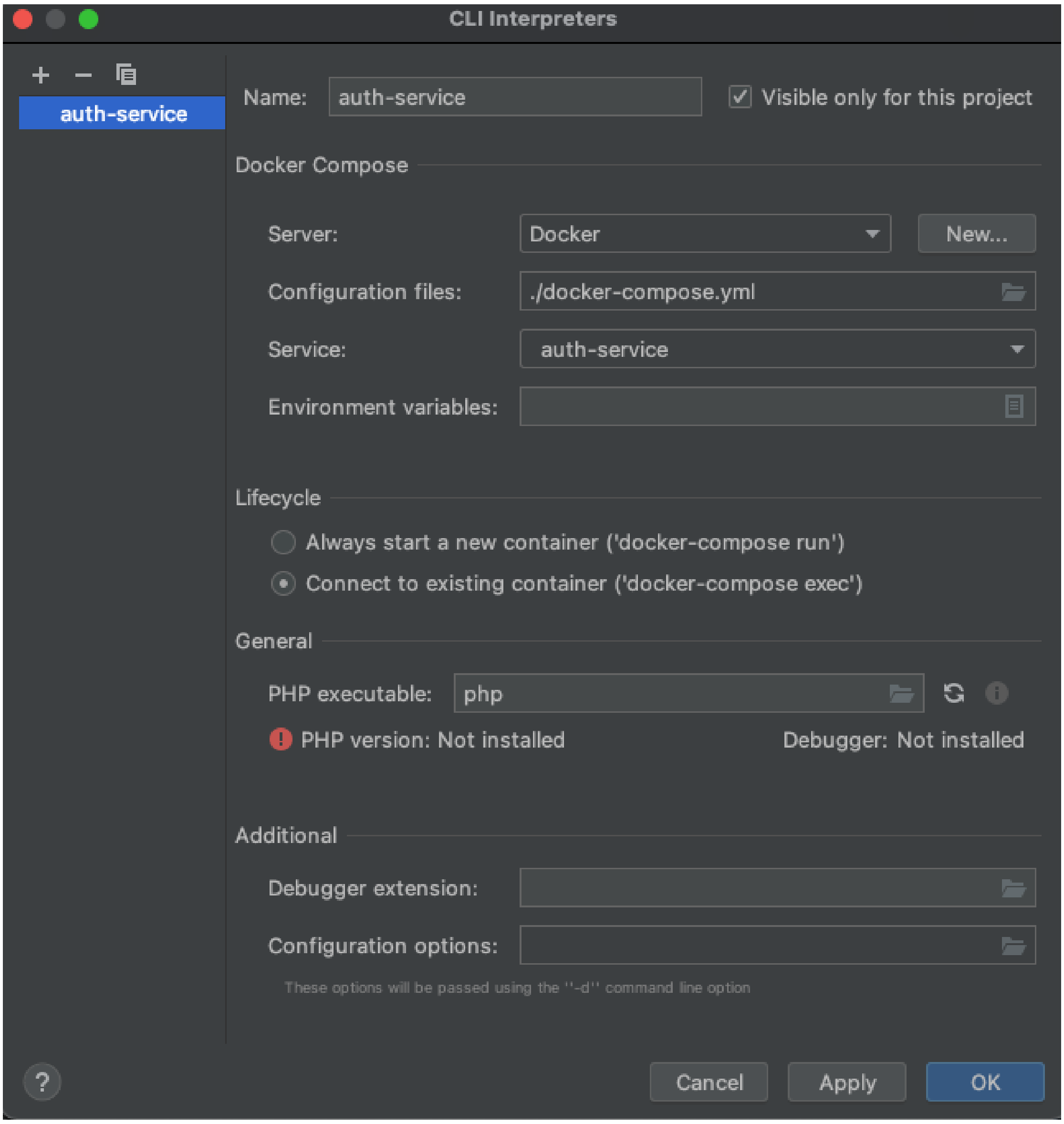
Certificates folder:

Path mappings:

Virtual machine path	Local path
/Users	/Users

Connection successful

After the new configuration, CLI Interpreters should look like this:
Set Lifecycle to "**Connect to existing container**"



CLI Interpreters

auth-service

Name: auth-service ☑ Visible only for this project

Docker Compose

Server: Docker

Configuration files: ./docker-compose.yml

Service: auth-service

Environment variables:

Lifecycle

○ Always start a new container ('docker-compose run')

● Connect to existing container ('docker-compose exec')

General

PHP executable: php

PHP version: Not installed Debugger: Not installed

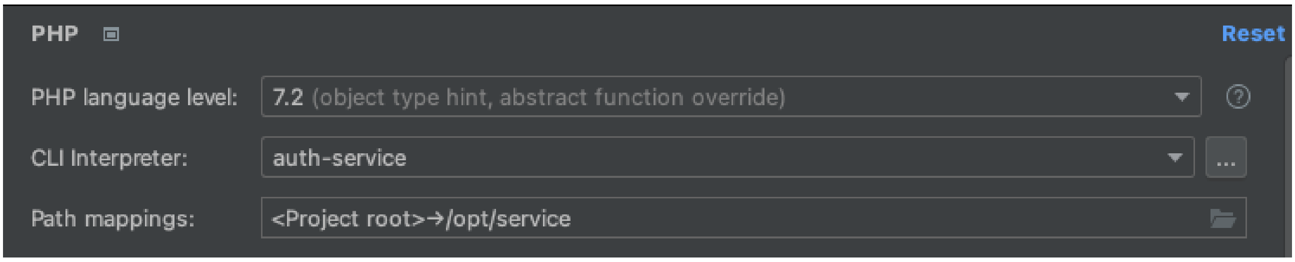
Additional

Debugger extension:

Configuration options:

These options will be passed using the ''-d'' command line option

PHP CLI



PHP

PHP language level: 7.2 (object type hint, abstract function override)

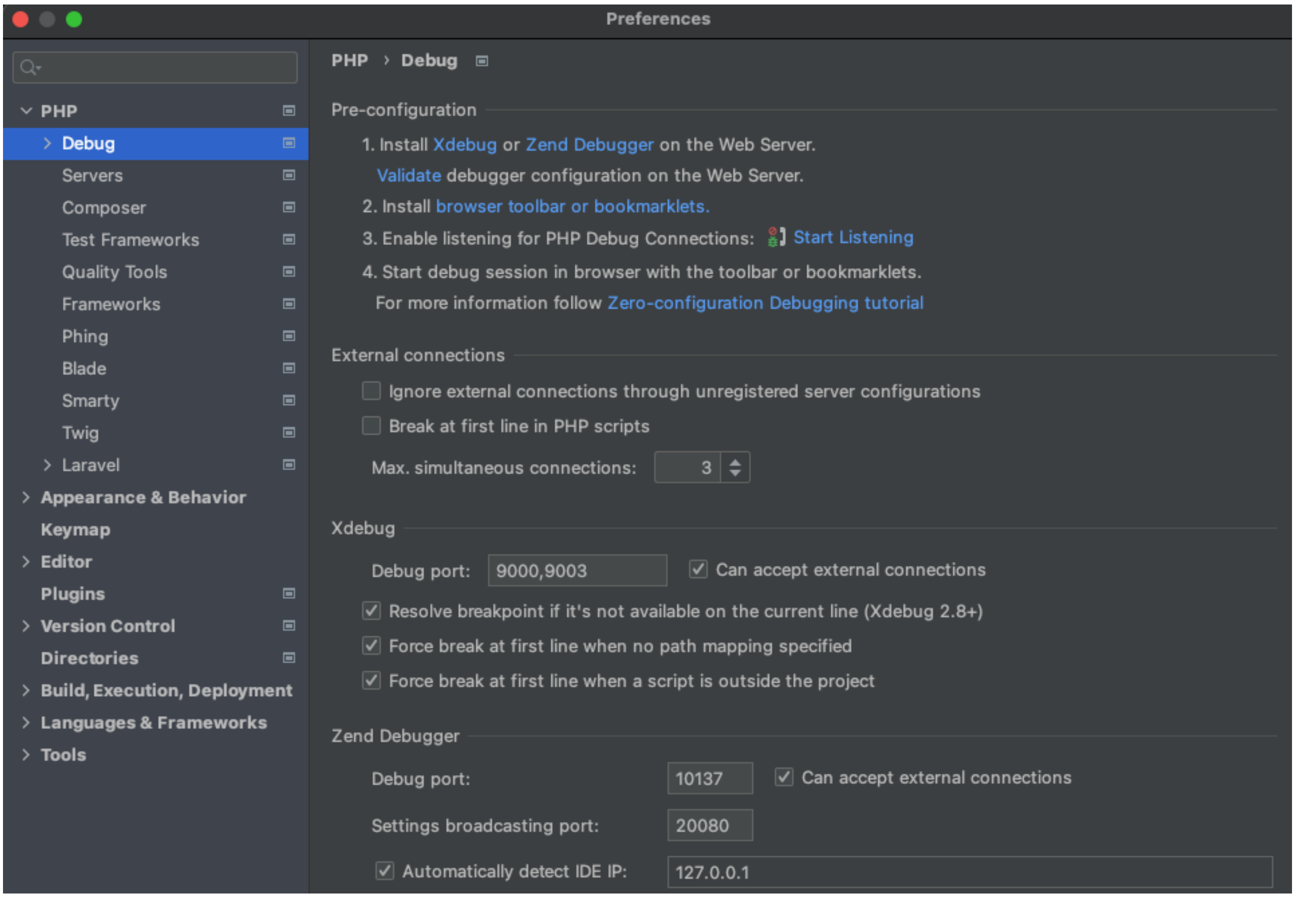
CLI Interpreter: auth-service

Path mappings: <Project root>→/opt/service

PHP Debug

Now we need to check Debug section ***Preferences -> PHP*** -> ***Debug***

Enable "Can accept external connection"
Xdebug Debug port: **XDEBUG_REMOTE_PORT**



PHP › Debug

Pre-configuration

1. Install Xdebug or Zend Debugger on the Web Server.
   Validate debugger configuration on the Web Server.
2. Install browser toolbar or bookmarklets.
3. Enable listening for PHP Debug Connections: Start Listening
4. Start debug session in browser with the toolbar or bookmarklets.
   For more information follow Zero-configuration Debugging tutorial

External connections

□ Ignore external connections through unregistered server configurations
□ Break at first line in PHP scripts
Max. simultaneous connections: 3

Xdebug

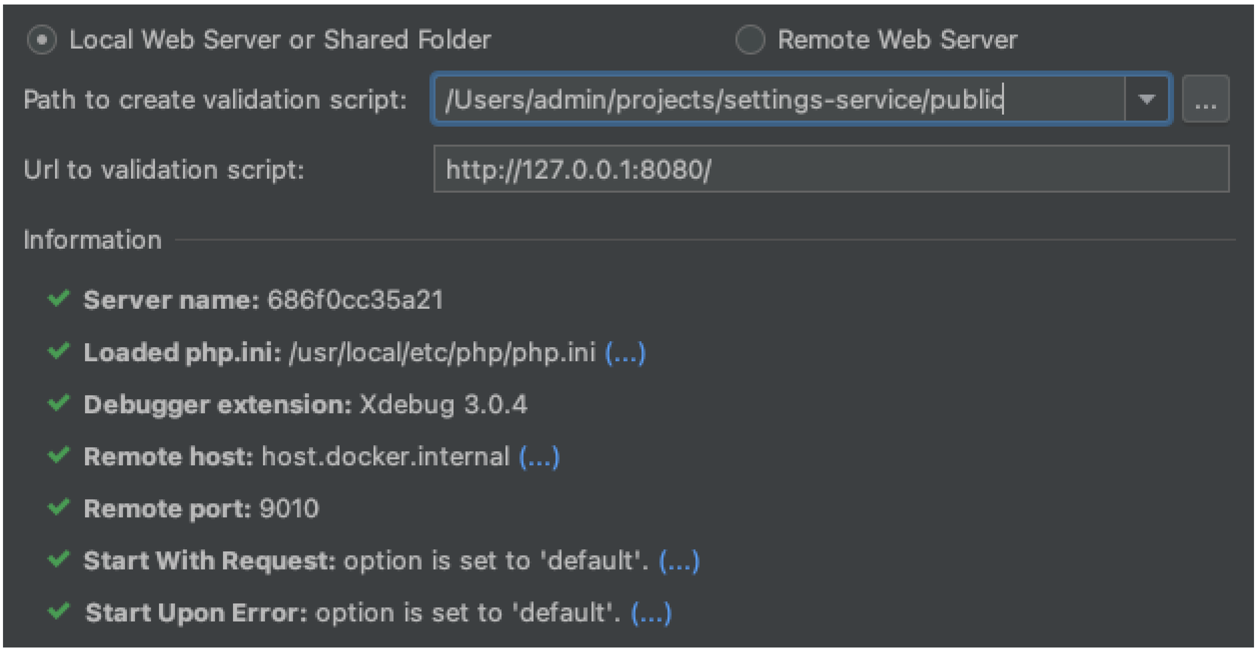
Debug port: 9000,9003 ☑ Can accept external connections
☑ Resolve breakpoint if it's not available on the current line (Xdebug 2.8+)
☑ Force break at first line when no path mapping specified
☑ Force break at first line when a script is outside the project

Zend Debugger

Debug port: 10137 ☑ Can accept external connections
Settings broadcasting port: 20080
☑ Automatically detect IDE IP: 127.0.0.1

Validate Xdebug

Go to ***Preferences -> PHP*** -> ***Debug,*** Pre-configuration click ***Validate***
Choose "**Local Web Server**"
Path: chose the local public folder of your project (*For Laravel or Lumen, set path to the public folder, not root*). Url is the same you use on the local machine, for example, http://127.0.0.1:8480/



● Local Web Server or Shared Folder ○ Remote Web Server

Path to create validation script: /Users/admin/projects/settings-service/public

Url to validation script: http://127.0.0.1:8080/

Information

✓ **Server name:** 686f0cc35a21

✓ **Loaded php.ini:** /usr/local/etc/php/php.ini (...)

✓ **Debugger extension:** Xdebug 3.0.4

✓ **Remote host:** host.docker.internal (...)

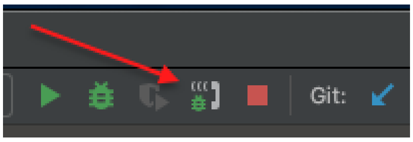
✓ **Remote port:** 9010

✓ **Start With Request:** option is set to 'default'. (...)

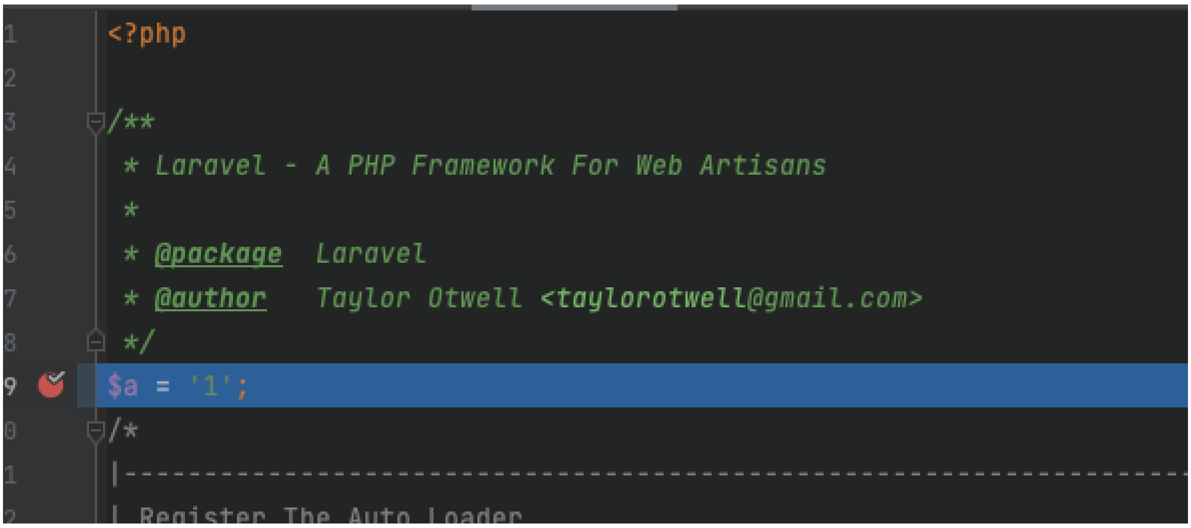
✓ **Start Upon Error:** option is set to 'default'. (...)

Enjoy your Xdebug experience

Click "**Start Listening for PHP Debug Connections**"



Set breakpoint:

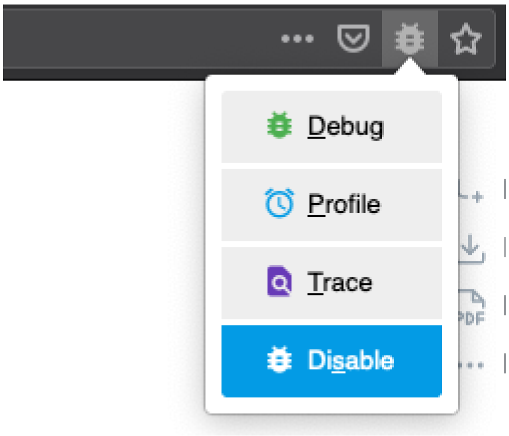


```
<?php

/**
 * Laravel - A PHP Framework For Web Artisans
 *
 * @package  Laravel
 * @author   Taylor Otwell <taylorotwell@gmail.com>
 */
$a = '1';
/*
|--------------------------------------------------------------------------
| Register The Auto Loader
```

And update the page on a browser.

In some case, you may need to install Browser extention


Debug
Profile
Trace
Disable