

PHP Guides

- [Set Up PHP Development Environment](#)
- [PHP Tooling Guide](#)
- [PHP Style Guide](#)
- [Laravel Development Guides](#)
 - [Bootstrapping Laravel Project](#)
 - [Laravel Tooling Guide](#)
 - [Authentication Support](#)
 - [Immutable Date Setup](#)
 - [Security Hardening Setup](#)
 - [Health Check Setup](#)
- [Makefile Snippets](#)

Set Up PHP Development Environment

www.php.net [documentation](#) [php/php-src](#)

Installation

```
sudo apt install php-cli php-curl php-mbstring php-mysql php-pdo php-sqlite3  
php-xml php-zip
```

Install Composer from Official Distribution

getcomposer.org [documentation](#) [downloads](#)

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') ===  
'c8b085408188070d5f52bcfe4ecfbee5f727afa458b2573b8eaaf77b3419b0bf2768dc67c86944  
da1544f06fa544fd47') { echo 'Installer verified'.PHP_EOL; } else { echo  
'Installer corrupt'.PHP_EOL; unlink('composer-setup.php'); exit(1); }"  
php composer-setup.php  
php -r "unlink('composer-setup.php');"
```

```
sudo chown root:root composer.phar  
sudo mv composer.phar /usr/local/bin/composer
```

Install Laravel Globally

laravel.com [documentation](#) [laravel/laravel](#)

```
composer global require laravel/installer
```

```
echo 'export PATH="$(composer global config bin-dir --absolute --quiet):$PATH"'  
>> ~/.bashrc
```

PHP Tooling Guide

EditorConfig Code Style

[EditorConfig Guide](#)

```
# SPDX-License-Identifier: 0BSD
# SPDX-FileCopyrightText: none

[* .html]
indent_size = 2
tab_width = 2

[* .js]
indent_size = 2
tab_width = 2
ij_continuation_indent_size = 4
ij_javascript_enforce_trailing_comma = whenmultiline

[* .php]
indent_size = 4
indent_style = space

[composer.json]
# Tooling uses 4-space indentation; match it.
indent_size = 4
indent_style = space
```

Exclude Files from Version Tracking

[Git Repository Guide](#)

```
# SPDX-License-Identifier: 0BSD
# SPDX-FileCopyrightText: none

/.phpunit.cache
```

```
/.phpunit.result.cache  
/vendor
```

PHP Style Guide

PHP Script Heading

```
<?php

/**
 * SPDX-License-Identifier: Apache-2.0
 * SPDX-FileCopyrightText: 2026 Tomáš Havlas <tomas@havlas.me>
 */
declare(strict_types=1);
```

Laravel Development Guides

Bootstrapping Laravel Project

[laravel.com documentation](https://laravel.com/documentation) [laravel/laravel](https://laravel.com/docs)

Create a New Laravel Application

```
laravel new --pest "[PROJECT_NAME]"
```

Create a New Laravel Package

```
laravel package "[PACKAGE_NAME]"
```

Laravel Tooling Guide

PHP Tooling Guide

Laravel Package Justfile

```
# SPDX-License-Identifier: 0BSD
# SPDX-FileCopyrightText: none
#
# Manage the PHP and Laravel toolchain of this package.

# Run the recipes at the project root, not at the `.just` directory.
set working-directory := '..'

HOST := "127.0.0.1"
PORT := "8000"

# Install the Composer dependencies.
install:
    composer install

# Serve the workbench application on the PHP development server.
dev host=HOST port=PORT:
    @php vendor/bin/testbench workbench:build --ansi
    @php vendor/bin/testbench serve --ansi --host="{{ host }}" --port="{{ port
}}"

# Run the Pest test suite.
test:
    composer test:unit
```

```
# Manage the PHP and Laravel toolchain of this package.
[group('laravel')]
mod laravel '.just/php-laravel.just'
```

Authentication Support

Installation

```
composer require laravel/sanctum  
composer require laravel/fortify
```

```
php artisan install:api  
php artisan vendor:publish --provider="Laravel\Sanctum\SanctumServiceProvider"  
php artisan config:publish cors
```

```
php artisan fortify:install  
php artisan vendor:publish --provider="Laravel\Fortify\FortifyServiceProvider"
```

Configuration

Immutable Date Setup

Configuration

```
// in app/Service/ImmutableDateService.php

use Carbon\CarbonImmutable;
use Illuminate\Support\Facades\Date;
use Illuminate\Support\ServiceProvider;

class ImmutableDateService extends ServiceProvider
{
    /**
     * Bootstrap any application services.
     */
    public function boot(): void
    {
        Date::use(CarbonImmutable::class);
    }
}
```

Security Hardening Setup

Configuration

```
// in app/Service/SecurityService.php

use Illuminate\Support\Facades\DB;
use Illuminate\Support\ServiceProvider;

class SecurityService extends ServiceProvider
{
    /**
     * Bootstrap any application services.
     */
    public function boot(): void
    {
        DB::prohibitDestructiveCommands($this->app->isProduction());
    }
}
```

Health Check Setup

[documentation spatie/laravel-health](#)

Installation

```
composer require spatie/laravel-health
```

```
php artisan vendor:publish --tag="health-config"  
php artisan vendor:publish --tag="health-migrations"
```

Configuration

```
// in app/Service/HealthCheckService.php  
  
use Illuminate\Support\ServiceProvider;  
use Spatie\Health\Checks\Checks\DatabaseCheck;  
use Spatie\Health\Checks\Checks\DebugModeCheck;  
use Spatie\Health\Checks\Checks\EnvironmentCheck;  
use Spatie\Health\Checks\Checks\UsedDiskSpaceCheck;  
use Spatie\Health\Facades\Health;  
  
class HealthCheckService extends ServiceProvider  
{  
    /**  
     * Bootstrap any application services.  
     */  
    public function boot(): void  
    {  
        Health::checks([  
            // https://spatie.be/docs/laravel-health/v1/available-  
checks/environment  
            EnvironmentCheck::new()->unless($this->app->environment('local')),  
            // https://spatie.be/docs/laravel-health/v1/available-checks/debug-  
mode  
            DebugModeCheck::new()->unless($this->app->environment('local')),  
            // https://spatie.be/docs/laravel-health/v1/available-checks/db-
```

```

connection
    DatabaseCheck::new(),
    // https://spatie.be/docs/laravel-health/v1/available-checks/used-
disk-space
    UsedDiskSpaceCheck::new(),
    // https://spatie.be/docs/laravel-health/v1/available-checks/queue
    // https://spatie.be/docs/laravel-health/v1/available-
checks/schedule
    ]);
}
}

```

```

// in routes/web.php

use Spatie\Health\Http\Controllers\HealthCheckResultsController;

Route::get('/up', HealthCheckResultsController::class);

```

```

// in routes/console.php
use Illuminate\Support\Facades\Schedule;

Schedule::command(\Spatie\Health\Commands\RunHealthChecksCommand::class)-
>everyMinute();

```

Custom Health Check Model

```

// in app/Model/HealthCheckResult.php

use Illuminate\Database\Eloquent\Attributes\Table;
use Spatie\Health\Models\HealthCheckResultHistoryItem;

#[Table(name: 'HEALTH_CHECK')]
class HealthCheckResult extends HealthCheckResultHistoryItem
{
    //
}

```

```

// in config/health.php

return [
    'result_stores' => [
        EloquentHealthResultStore::class => [
            'model' => HealthCheckResult::class,
        ],
    ],
];

```

Makefile Snippets

Laravel Based Project

Install Dependencies

```
.PHONY: local/install
local/install:
    composer install
```

Update Dependencies

```
.PHONY: local/update
local/update:
    composer update
```

Run Local Instance

```
DEVHOST := 127.27.0.1
DEVPORT := 8081

.PHONY: local/up
local/up:
    php artisan serve --host="$(DEVHOST)" --port="$(DEVPORT)"
```

Clean Wrokspace

```
.PHONY: clean
clean:
    -rm .phpunit.cache
    -rm .phpunit.result.cache

.PHONY: distclean
distclean: clean
    git clean --force -Xd
```