

Laravel 8.83 to 9.52.15 Upgrade Documentation

This document outlines the detailed steps required to upgrade a Laravel project from version 7.30 to 8.83. The upgrade includes updates to the composer.json file, adjustments to key package versions, and modifications to core files like Handler.php for compatibility with Laravel 8.83.

Approximate time required for this upgrade: 10-16 hours

Why Laravel 9.52.15 and Not Laravel 9.*?

We have specifically chosen Laravel 8.83 and not the latest 8.* series because Laravel 8.83 is the final release of the Laravel 8.x line and receives long-term support (LTS). It includes all critical bug fixes, security updates, and performance improvements. Choosing this version ensures maximum compatibility with our current package ecosystem such as altek/accountant, arcanedev/log-viewer, and others. By locking to this specific version, we avoid potential issues from newer 8.* releases and maintain a secure, stable codebase in production.

1. Backup the Project and Database

Backup the code: `cp -r laravel-adminpanel laravel-adminpanel-backup`

Or use Git:

```
git add .  
git commit -m "Backup before Laravel 9 upgrade"  
git tag pre-laravel-9  
git push origin dev-upgrade-8-to-9 --tags
```

Backup the database:

For MySQL: `mysqldump -u youruser -p yourdatabase > db_backup.sql`

2. Update Laravel Version in composer.json

Edit composer.json: `"laravel/framework": "^9.52.15"`

Also, verify compatibility of any Laravel-specific packages with Laravel 9.

3. Try to Update Composer Packages

Run: **composer update**

If you encounter memory errors: **COMPOSER_MEMORY_LIMIT=-1 composer update**

4. Fix Package Compatibility Issues

Problem 1: Package X requires laravel/framework ^8.83, found ^9.52.15

Solution: composer require vendor/package:^X.Y

Problem 2: Abandoned packages or those not compatible with Laravel 9.

Solutions:

- Check the GitHub repo for Laravel 9 support
- Replace it with an alternative package
- Temporarily remove it from composer.json

Then retry: **composer update**

5. Clear and Optimize Laravel Setup

After a successful update:

php artisan config:clear

php artisan route:clear

php artisan view:clear

php artisan cache:clear

composer dump-autoload

php artisan optimize

6. Handle Code Changes (e.g., Handler.php)

Laravel 9 introduces changes to core files like app/Exceptions/Handler.php.

- Adjust method signatures
- Remove deprecated methods
- Add missing imports

7. Check Project Functionality

Start the server: **php artisan serve**

Verify:

- All pages load correctly
- Forms and submissions work
- Admin panel, login/logout, dashboard
- All JS and CSS assets load properly
- No console or network errors

8. Run Migrations and Test Data

Run migrations: **php artisan migrate**

If needed: **php artisan db:seed**

9. Run Tests (If Available)

php artisan test

Fix any failing tests.

10. Final Cleanup

php artisan config:cache

php artisan route:cache

php artisan view:cache

If using GrumPHP: **vendor/bin/php-cs-fixer fix**

Commit without GrumPHP check (if needed):

git commit --no-verify -m "Laravel upgraded to 9.52.15"

Summary of Commands

Backup

```
cp -r laravel-adminpanel laravel-adminpanel-backup  
mysqldump -u root -p dbname > backup.sql
```

Update

```
composer update  
COMPOSER_MEMORY_LIMIT=-1 composer update
```

Fix packages

```
composer require vendor/package:^version
```

Clear + optimize

```
php artisan config:clear  
php artisan optimize  
php artisan migrate  
php artisan serve  
php artisan test
```

Commit

```
git commit -am "Upgraded Laravel to 9.52.15"  
git push
```

Composer Package Changes

1. Updated Packages:

PHP Version:

Old Version: ^7.4

New Version: ^8.0

Reason for Change: Laravel 9.x officially supports PHP 8.0 and higher, which brings improved performance, better error handling, and new features. It's necessary to upgrade PHP for better compatibility.

altek/accountant:

Old Version: ^2.1

New Version: ^3.3

Reason for Change: Upgraded to version 3.3 for compatibility with Laravel 9.x. The previous version 2.1 caused issues with newer Laravel versions.

arcanedev/log-viewer:

Old Version: ^7.0

New Version: ^8.0

Reason for Change: Upgraded to 8.0 to resolve compatibility issues with Laravel 9.x. The newer version supports the changes introduced in Laravel 9.x.

laravel/framework:

Old Version: ^7.30

New Version: ^8.83

Reason for Change: Upgraded to Laravel 8.83 to ensure compatibility with the latest features and improvements in the Laravel ecosystem.

nunomaduro/collision:

Old Version: ^4.0

New Version: ^5.0

Reason for Change: Upgraded to version 5.0 to ensure compatibility with Laravel 9.x. This version provides improved error handling and UI improvements.

2. Removed Packages:

facade/ignition:

Removed Versions: ^1.11

Reason for Removal: Replaced by nunomaduro/collision, which is now the default error handler in Laravel 9.x.

phpro/grumphp:

Removed Versions: ^1.0

Reason for Removal: This package is no longer compatible with Laravel 9.x and has been replaced by more suitable alternatives.

roave/security-advisories:

Removed Versions: dev-master

Reason for Removal: Laravel now includes built-in dependency management and security advisory tools, making this package redundant.

Conclusion:

This upgrade process ensures that your project is fully compatible with **Laravel 9.52.15**, while also updating necessary packages and removing deprecated or unnecessary ones. By following these steps, you can leverage the new features, improvements, and better performance provided by **Laravel 9.52.15**, ensuring enhanced **security**, **maintainability**, and a smoother development experience.