Laravel 9.52.15 to 10.47 Upgrade Documentation

This document outlines the detailed steps required to upgrade a Laravel project from version 9.52.15 to 10.47 The upgrade includes updates to the composer.json file, adjustments to key package versions, and modifications to Handler.php to ensure compatibility with Laravel 10.47.

Approximate time required for this upgrade: 8-13 hours

Why Laravel 10.47 and Not Laravel 10.*?

We have specifically chosen Laravel 10.47 instead of using a wildcard version like `^10.0` because Laravel 10.47 is the most stable and up-to-date release in the Laravel 10 series at the time of upgrade. It includes all relevant bug fixes, security patches, and performance enhancements made available through the Laravel 10 lifecycle up to that point.

By locking the version to 10.47 using `"laravel/framework": "^10.47"`, we ensure consistent application behavior, reduce the risk of encountering issues from earlier 10.x releases, and maintain compatibility with other core dependencies such as `altek/accountant`, `arcanedev/log-viewer`, and others

Laravel 10 is an LTS (Long-Term Support) release, which means it will receive bug fixes for 2 years and security updates for 3 years. Choosing the latest patch within an LTS release not only aligns with best practices but also guarantees the most secure and optimized version of the framework for long-term production use. This strategy minimizes upgrade risks, simplifies future maintenance, and provides a solid, dependable foundation for the application.

---

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-9-to-10 --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": "^10.47"**
Also, verify compatibility of any Laravel-specific packages with Laravel 7.

---

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 ^9.52.15, found ^10.47
Solution: composer require vendor/package:^X.Y

Problem 2: Abandoned packages or those not compatible with Laravel 10.
Solutions:

- Check the GitHub repo for Laravel 10 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, run the following commands to clear and optimize the Laravel setup:

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 10 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

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: git commit **--no-verify** -m "Laravel upgraded to 10.47"

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 10.47"
  git push

Composer Package Changes:

1. Updated Packages:

PHP Version:
Old Version: `^7.4`
New Version: `^8.1`

Reason for Change: Laravel 10 requires PHP 8.1 or newer to take advantage of native type declarations, better performance, enum support, and improved error handling. Upgrading ensures compatibility and future-proofing of the application

arcanedev/log-viewer:
Old Version: `^9.0`
New Version: `^10.0`
Reason for Change: The older version was not compatible with Laravel 10.x. Upgrading to version 10 ensures compatibility with the Laravel 10 framework and brings improvements in performance, bug fixes, and support for updated dependencies.

knuckleswtf/scribe:
Old Version: `^3.22`
New Version: `^4.0`
Reason for Change: To align with Laravel 10 compatibility requirements. Version 4.0 introduces enhanced OpenAPI support, improved generation of API docs, and better handling of annotations and route changes.

laravel/framework:
Old Version: `^9.52.15`

New Version: `^10.47`

Reason for Change: Laravel 10 includes long-term support (LTS), PHP 8.1+ support, and new features like native type declarations and more consistent exception handling. This upgrade ensures future compatibility, better performance, and security improvements.

`laravel/passport:`

Old Version: `^10.0`

New Version: `^11.0`

Reason for Change: To maintain compatibility with Laravel 10. The newer Passport version includes important security patches and support for changes in Laravel 10's authorization components.

`laravel/ui:`

Old Version: `^3.2`

New Version: `^4.0`

Reason for Change: Required to support Laravel 10. The newer version provides updated UI scaffolding for Vue/React/Bootstrap that aligns with the latest frontend tooling and Laravel changes.

`symfony/console:`

Old Version: `^6.0.9`

New Version: `^6.2`

Reason for Change: Symfony Console is a key component used by Laravel's Artisan command-line tool. Upgrading ensures compatibility with Laravel 10's internal dependencies and provides performance and usability enhancements.

`yajra/laravel-datatables-oracle:`

Old Version: `^9.10`

New Version: `^10.0`

Reason for Change: Version 10 is compatible with Laravel 10 and PHP 8.1+. The update includes bug fixes, improved compatibility with DataTables JS, and cleaner API handling.

3. Removed Packages:

`albertcht/invisible-recaptcha:`

Removed Version: `^1.9`

Reason for Change: This package is no longer actively maintained and does not support Laravel 10. Alternative methods (such as Google's official reCAPTCHA or other maintained packages) are recommended for spam protection in modern Laravel versions.

fideloper/proxy:
> Removed Version**: ^4.0**

Reason for Change: Starting with Laravel 9, trusted proxy handling has been moved into the framework itself using `Laravel\Http\Middleware\TrustProxies`. The fideloper/proxy package is deprecated and unnecessary in Laravel 10.

Conclusion:

This upgrade process ensures that your project is fully compatible with **Laravel 10.47**, while also updating necessary packages and removing deprecated or unsupported ones. By following these steps, you can leverage the new features, performance improvements, and long-term support (LTS) provided by Laravel 10.47, ensuring enhanced security, improved maintainability, and a more modern and efficient development experience.