**🚀 Guida al Deploy di Progetti Laravel su Netsons**

**🧹 Pulizia del progetto prima del deploy**

Prima di comprimere il progetto:

php artisan cache:clear

php artisan config:clear

php artisan route:clear

php artisan view:clear

npm run build

✅ Questo assicura che il progetto sia pulito e pronto per la produzione.

**📦 Compressione e caricamento**

1. **Comprimere il progetto in formato .zip**
2. **Caricare il file ZIP nella directory**  
   home/utente/nome\_progetto  
   *(sostituisci utente e nome\_progetto con i tuoi dati)*
3. **Estrarre il contenuto del file ZIP**

**🗂️ Organizzazione delle cartelle**

1. Crea la cartella:  
   home/utente/public\_html/nome\_progetto
2. Sposta **tutti i file** da:  
   home/utente/nome\_progetto/public  
   👉 a 👉  
   home/utente/public\_html/nome\_progetto
3. Nella cartella originale public, lascia solo:
4. build/

└── manifest.json

**🛠️ Modifica del file** [**index.php**](https://index.php)

📍 Percorso: home/utente/public\_html/nome\_progetto/[index.php](https://index.php)

Sostituisci il contenuto con:

<?php

use Illuminate\Foundation\Application;

use Illuminate\Http\Request;

define('LARAVEL\_START', microtime(true));

*// Determine if the application is in maintenance mode...*

if (file\_exists($maintenance = '/home/apghicha/nome\_progetto/storage/framework/maintenance.php')) {

require $maintenance;

}

*// Register the Composer autoloader...*

require '/home/apghicha/nome\_progetto/vendor/autoload.php';

*// Bootstrap Laravel and handle the request...*

*/\*\* @var Application $app \*/*

$app = require\_once '/home/apghicha/nome\_progetto/bootstrap/app.php';

$app->handleRequest(Request::capture());

🔁 Ricorda di sostituire apghica e nome\_progetto con i tuoi dati reali.

**🔐 Configurazione del database**

Modifica il file .env con i dati di accesso al database forniti da Netsons:

DB\_CONNECTION=mysql

DB\_HOST=localhost

DB\_PORT=3306

DB\_DATABASE=nome\_database

DB\_USERNAME=nome\_utente

DB\_PASSWORD=la\_tua\_password

**🎨 Aggiornamento CSS/JS senza ricaricare tutto**

Se hai modificato solo i file CSS o JS:

1. Esegui di nuovo:

npm run build

1. Carica i file generati da:

local/public/build/assets/

👉 a 👉

public\_html/nome\_progetto/build/assets/

1. Carica anche il file [manifest.json](https://manifest.json) in:

home/utente/nome\_progetto/public/build/

📎 In alternativa, puoi copiare direttamente le nuove righe di CSS/JS nel file manager online.

**📁 Gestione contenuti multimediali senza storage**

Per caricare manualmente immagini o video senza utilizzare lo storage Laravel, segui questi passaggi:

**🔧 1. Crea la cartella per i file**

Organizza i contenuti multimediali in una directory accessibile pubblicamente:

public/build/assets/img

public/build/assets/video

Puoi usare anche una sola cartella img se preferisci.

**⚙️ 2. Configura il filesystem**

Modifica il file config/[filesystems.php](https://filesystems.php) per definire un nuovo disco public che punti alla cartella appena creata:

        'public' => [

            'driver' => 'local',

            'root' => storage\_path('build/assets'),

            'url' => env('APP\_URL').'/build/assets',

            'visibility' => 'public',

            'throw' => false,

            'report' => false,

        ],

**🔗 3. Caricamento manuale**

I file multimediali (immagini, video, ecc.) devono essere caricati manualmente nella cartella build/assets. Non verranno gestiti tramite lo storage Laravel.