

Laravel è un Framework.

Cos'è un framework?

E' un insieme di librerie, strumenti e linee guida che consentono di sviluppare software in modo efficiente, standardizzato e strutturato.

->Cartella per la connessione al database (.php)

-> Cartella che gestisce il codice delle richieste http (.php)

Composer ->

Dependency Manager -> Gestore di dipendenze, Gestore di pacchetti

Dipendenze: sono tutte le librerie di cui ha bisogno Laravel per funzionare

Una dipendenza è una libreria che serve a Laravel per funzionare

Per installare Laravel

1. Installare Composer
2. Istruzione per installare un nuovo progetto in Laravel

```
laravel new mioprogetto
```

3. Devo fare il run del web server:

```
php artisan serve
```

/vendor -> tutte le librerie che servono a Laravel per funzionare e che ci forniscono funzionalità che non dobbiamo scrivere

composer.json -> è un file che contiene una descrizione di tutte le dipendenze di cui il progetto ha bisogno

composer.lock -> elenca tutte le dipendenze e le loro versioni esatte che sono state scaricate e installate nel progetto insieme alle loro dipendenze. Contiene anche le info per l'aggiornamento
.env -> il file di configurazione delle variabili di ambiente del mio progetto (non viene pushato)

.env.example -> viene pushato e contiene tutte le variabili che il mio progetto deve poter usare ma con valorizzazioni di esempio

.gitignore -> è un file che elenca tutte le cartelle e i file che non vogliamo pushare

.artisan -> l'interfaccia a riga di comando che laravel ci mette a disposizione per impartire i comandi al progetto

package.json -> contiene le dipendenze lato js

/app -> contiene i file in cui io vado a modellare la logica della mia applicazione, vado a definire il dominio dell'applicazione

/database -> gestisco tutto ciò che riguarda il database

/public -> è la document root -> posso vedere dal browser solo quello che è contenuto in questa cartella

/config -> cartella in cui sono contenuti i file di configurazione dei vari pacchetti e dell'applicazione stessa

/resources -> tutti i file che gestiscono la parte visuale: html, css, javascript, le viste

/routes -> qui dentro gestiamo le richieste HTTP

/test -> php unit test

/storage -> contiene tutti i file temporanei dell'applicazione come i log, le sessioni, le cache delle viste...Ed in più contiene anche i file come immagini, pdf, audio, video del nostro progetto

/bootstrap -> si occupa di inizializzare l'applicazione

Errori

View [giggi] not found.

E' l'errore che si genera quando ritorno una vista che non esiste