Laravel en testen maken

Voor het schrijven van testen in Laravel heb je vanuit de basisinstallatie niets nodig. Enkel een werkende Laravel installatie.

We gebruiken voor onze testcase een voorbeeld class die we nog gaan maken.

Er zit nog geen code in maar we gaan eerst kijken of de test en testroutines werken.

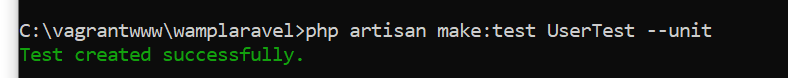
Wat hebben we hiervoor nodig:

* Een installatie Laravel
* Command line
* IDE editor: phpStorm of Visual code studio

Open je command terminal op je projecten folder:

In mijn voorbeeld draait Laravel in folder: c:\vagrantwww\wamplaravel

Voer hier de volgende code in:



Artisan heeft voor je een voorbeeld class gemaakt.

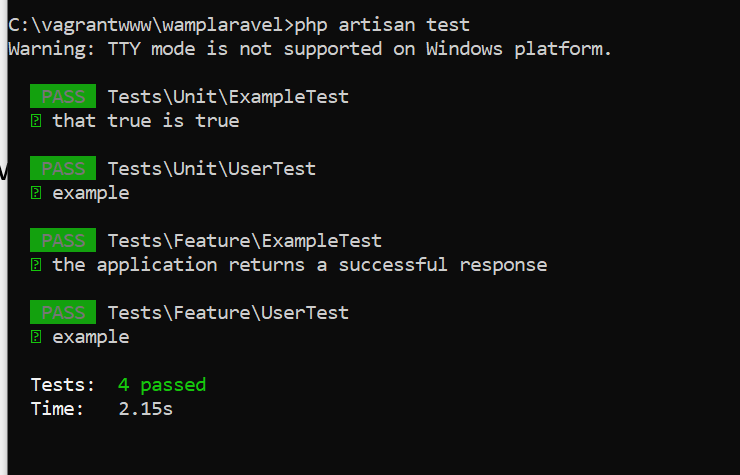
Deze gaan we dus verder gebruiken en aanvullen met nieuwe tests.

Open je IDE en kijk in folder “tests” --> “Unit” 🡪 UserTest.php

Hoe voer je een test uit?

Zo ziet het er uit als je nu een test uitvoert.

Er zijn nu 4 testen uitgevoerd.



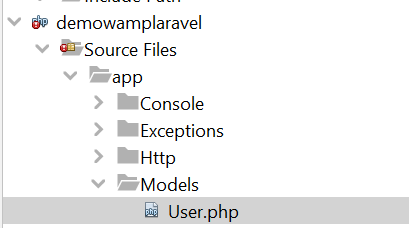
Een nieuwe test schrijven?

In onze modelclass gaan we een echte code toevoegen die we verder met code uitwerken zoals het hoort.

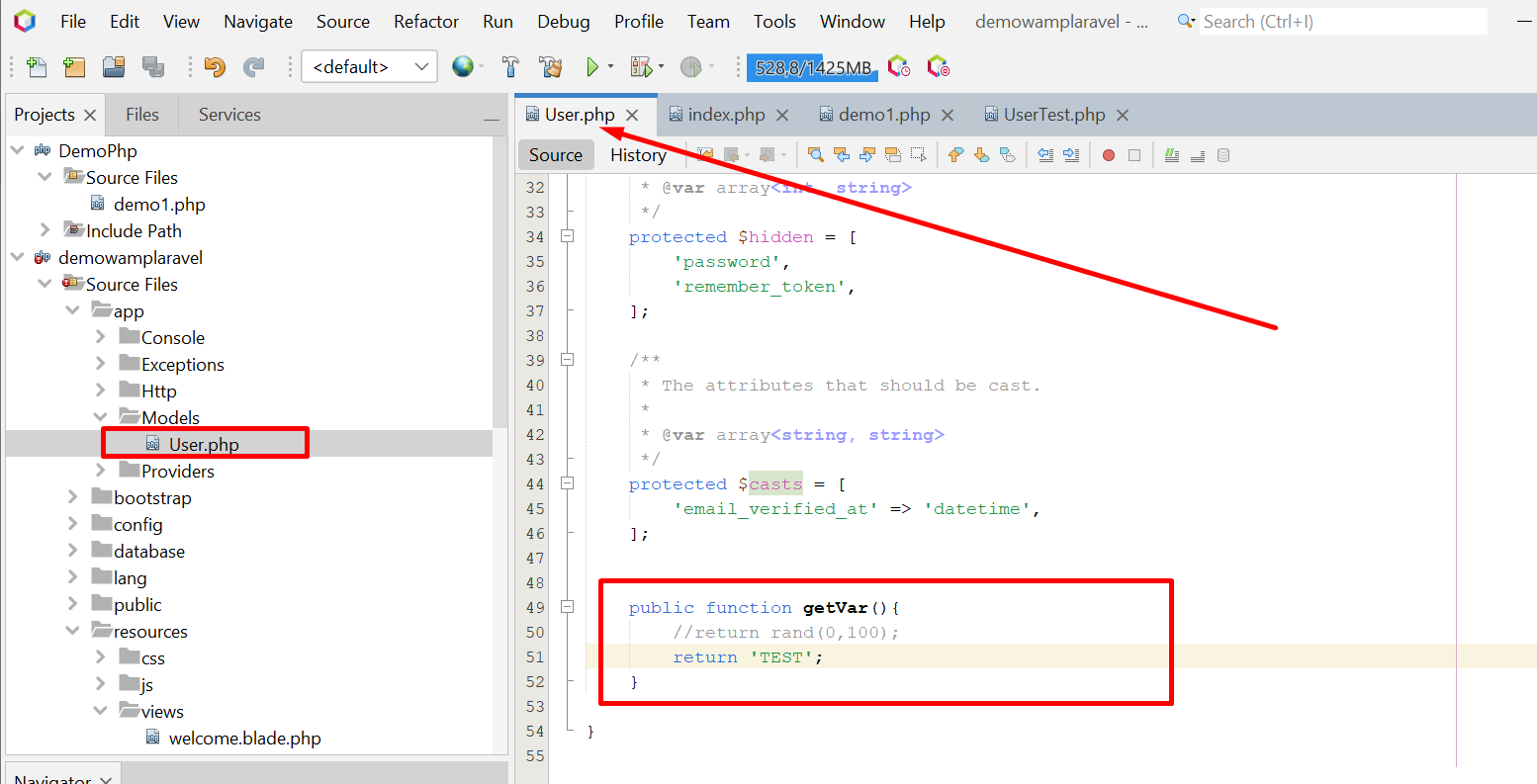
We maken hier een functie in en de functie zal een integer waarde geven.

Function getVar() levert een random getal en wij gaan hier een test op schrijven om te controleren of de functie getVar() ook een echte integer uitstuurt.

Wat is de nut hiervan? Kan handig zijn omdat je soms na een berekening of routine altijd een integer waarde wilt hebben.



Voeg de functie getVar toe aan je code:



Het is een kleine functie, doet verder weinig maar we gaan hier een test voor schrijven.

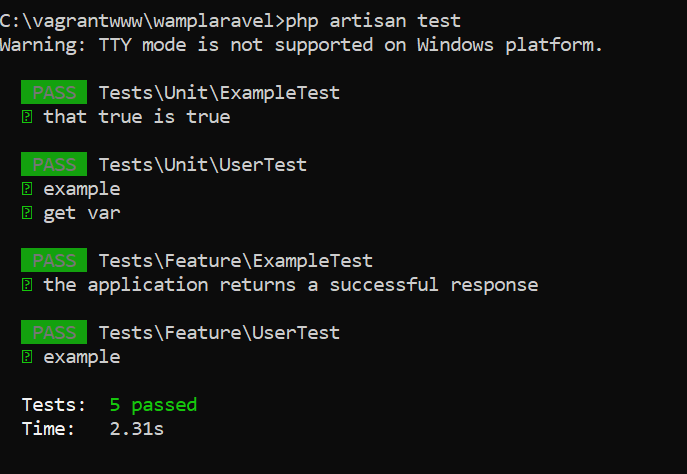
Wat weten we? getVar geeft een integer waarde terug (feit).

Test? We willen een test maken om dit te controleren.

Dus we schrijven een test die checkt of de waarde uit de functie getVar ook een integer is.

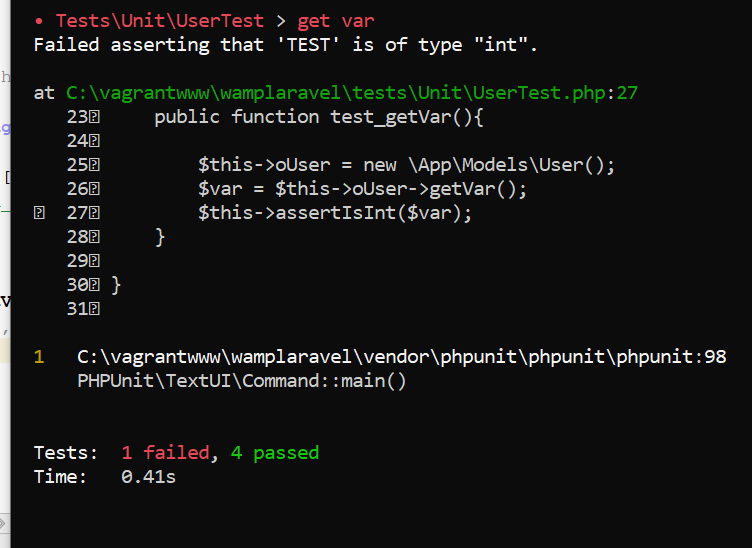
We gaan weer de testen uitvoeren: onderstaande zie je dat alles goed gaat.

Er zijn 5 testen uitgevoerd, en groen is altijd goed.



Wanneer het fout gaat endus de test niet slaagt dan krijg je dus een foutmelding.

Je moet dus deze fout fixen want de code (jouw test) geeft een fout aan.



Later in de lessen gaan we nader in op de test en testroutines.

EINDE LES TESTEN