|  |  |
| --- | --- |
|  |  |
| **Project 7 Laravel** |  |
|  |  |
|  |  |

**Les 4 - Templates**

We gaan het vandaag hebben over **Templates / Views**

1. Wat is het?
2. Waarom gebruiken we het?
3. Voorbereiden met de opdrachten naar een eigen crud

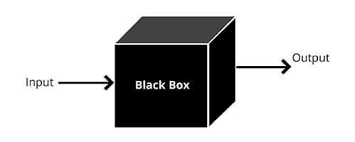
## Hoe deden we het hiervoor in PHP?

*(klassikaal bespreken)*

Voordelen?

Nadelen?

## Wat is een template (lees: views)?



Een template is eigenlijk alles wat zich bezig houdt met het eindresultaat van een PHP afhandeling.

Zie bovenstaande afbeelding: “output”. Binnen Laravel noemen we dit ook ‘views’.

Meestal is dit ook de laatste fase waarin een html pagina getoond wordt.

Grondregel van een template/view:

***Op het moment van een output beschikt de template-engine over alle data die nodig is om een bepaalde output te maken.***

Pas op de laatste stap wordt er een “echo” uitgevoerd. Dit was voorheen natuurlijk anders met losse php bestanden; tussentijdse echo’s leverden vaak problemen op…

Dit wordt automatisch zichtbaar gemaakt worden door een visuele ‘webpagina’ die je ook toont.

Dus je hoeft zelf niet nog een keer een echo uit te voeren. Op het moment dat je gebruik maakt van de template engine gebeurt er automatisch een output.

De kracht van een template engine is ook om data dusdanig te visualiseren dat je er alle kanten mee op kunt qua design. Alles voor designers!

Loop je tegen beperkingen aan, dan zul je het een stap daarvoor moeten oplossen.

In een team kun je dus al van twee kanten starten; 1 designs uitwerken tot werkbare templates en andere team houdt zich met de backend ontwikkeling bezig. Dit alles zonder dat ze elkaar storen met bijvoorbeeld foutmeldingen.

Handig hè 😊

## Soorten templates

Er zijn verschillende soorten templates die we willen gebruiken.

Wanneer we een applicatie maken met een front-end gedeelte maar ook

Een admin-gedeelte dan hebben we vaak te maken met 2 hoofdtemplates.

In grote lijnen zijn ze van elkaar dusdanig gescheiden dat ze elkaar niet mogen beïnvloeden.

Daarbinnen hebben ze eigen subtemplates die een verdere lay-out verzorgen.

* Structuur is heel erg belangrijk
* We hebben altijd 1 template die we zoveel mogelijk herbruiken (voorkom kopieën)

Hoe efficiënter de opzet van de templates is bedacht, hoe makkelijker dit is om voor een frontender de applicatie verder visueel aan te kleden.

Een een grote kracht is ook dat je kan samenwerken met designers die inhoudelijk niet met de applicatie bemoeien, maar puur visueel kunnen bezig zijn.

Let op:

* Vanuit de templates gaan we nooit en te nimmer meer extra data vanuit de templates proberen te halen.

Wat wordt hiermee bedoeld? Je templates is als het ware je eindstation; je moet het er maar mee doen met de data die je vanuit de route (lees: controllers) verder aangeleverd krijgt. Als je met een crud lijst bezig bent om bijvoorbeeld een listing (weergave van records) te maken en je krijgt dit niet via de route (lees: controllers) aangeleverd, dan is het niet de bedoeling om in je templates een find:all() te doen. Ook al is de uitwerking nog zo eenvoudig, het gaat om de principe dat de templates een taak hebben om de data te visualiseren (eenrichtingsverkeer). Opmaak mag natuurlijk altijd en in elk vorm, en je zult zien dat een template engine je hier enorm bij helpt.

De template engine heeft zelf ook een eigen programmeer-set aan functies.

Zo kun je condities opgeven: als dit dan dat (if /else)

Maar ook herhalen van routines is ook mogelijk.

Meestal krijg je vanuit PHP een resultaat-set en is in de meeste gevallen een array (objecten doorgeven kan natuurlijk ook).

Het ‘loopen’ van zo’n array kun je gemakkelijk een template doorheen halen.

Meestal zijn de crud acties vrij algemeen en kun je deze vaak herbruiken voor bijna elke tabel.

Een stuk filtering is ook mogelijk. Bijvoorbeeld tekst naar kleine letters, of van de tekst eerste karakter een hoofdletter, of tekst zonder spaties.

We gaan niet door alle mogelijkheden heen binnen de template engine. Hier is het lesvak te kort helaas.

Als je de voorgenoemde template-functies beheerst dan kun je ook al mooie dingen maken 😊

GA NAAR DE OPDRACHTEN