



Orbit

Onderzoek

Wat zijn de mogelijk herbruikbare
functionaliteiten vanuit Laravel Horizon?

Naar	Luuk de Weijer
	Scrumble
Betreft	Orbit implementatie
Auteur(s)	Servi Huijbregts
Email	servi@scrumble.nl

Inleiding en doelstelling

Inleiding Orbit is een applicatie die ervoor zorgt dat er meer inzicht komt in de verwerking van jobs en queues. Deze applicatie moet gemaakt worden omdat het huidige systeem dat dit faciliteert, Laravel Horizon, niet toereikend en gebruiksvriendelijk genoeg is.

Doelstelling Het doel is om een eigen variant van Laravel Horizon te ontwikkelen die wel het inzicht en de gebruiksvriendelijkheid biedt wat de developers van Scrumble vereisen. Het is echter nog niet bekend wat de einddoelstellingen van het project nou precies inhouden, dit moet nog verder onderzocht/aangevuld worden als de mogelijkheden van het project verder onderzocht zijn.

Onderzoeksvragen

Hoofdvraag Wat zijn de mogelijk herbruikbare functionaliteiten vanuit Laravel Horizon?

Deelvragen Welke functionaliteiten biedt Laravel Horizon momenteel al?
Welke hiervan kan ik eventueel hergebruiken?
Welke hiervan kan ik uitbreiden om eventuele requirements te behalen?

Deelvraag 1 & 2

Vraag Welke functionaliteiten biedt Laravel Horizon momenteel al? Welke hiervan kan ik eventueel hergebruiken?

Resultaat Laravel Horizon biedt een breed scala aan functionaliteiten aan, ik zal deze kort opsommen in een tabel.

Route	Type	Controller	Description
/stats	GET	DashboardStatsController@index	Retrieves statistics for the dashboard
/workload	GET	WorkloadController@index	Shows the current worker's workload
/masters	GET	MasterSupervisorController@index	Shows the list of all masters
/monitoring	GET	MonitoringController@index	Shows the monitoring page
/monitoring	POST	MonitoringController@store	Store a new monitoring entry
/monitoring/{tag}	GET	MonitoringController@paginate	Paginate monitoring entries by tag
/monitoring/{tag}	DELETE	MonitoringController@destroy	Delete monitoring entries by tag

/metrics/jobs	GET	JobMetricsController@index	Shows job metrics
/metrics/jobs/{id}	GET	JobMetricsController@show	Shows metrics for a specific job
/metrics/queues	GET	QueueMetricsController@index	Shows queue metrics
/metrics/queues/{id}	GET	QueueMetricsController@show	Shows metrics for a specific queue
/batches	GET	BatchesController@index	Shows batches of jobs
/batches/{id}	GET	BatchesController@show	Shows details of a specific batch
/batches/retry/{id}	POST	BatchesController@retry	Retry a specific batch
/jobs/pending	GET	PendingJobsController@index	Shows pending jobs
/jobs/completed	GET	CompletedJobsController@index	Shows completed jobs
/jobs/silenced	GET	SilencedJobsController@index	Shows silenced jobs
/jobs/failed	GET	FailedJobsController@index	Shows failed jobs
/jobs/failed/{id}	GET	FailedJobsController@show	Shows details of a specific failed job
/jobs/retry/{id}	POST	RetryController@store	Retry a specific failed job
/jobs/{id}	GET	JobsController@show	Shows details of a specific job.

Hiervan kan ik er natuurlijk veel gebruiken om de requirements te behalen.

Als ik de functionele requirements op een rij zet kan ik aangeven welke functies ik vanuit Laravel Horizon kan gebruiken om deze te realiseren:

Requirement	Description	Functie Horizon
Applicatie aanmaken en beheren	Een formulier met de mogelijkheid om een naam en een unieke identifier in te stellen om gemakkelijk te zoeken naar applicaties binnen de front-end van de tool.	N.V.T.
Alerts instellen	Een functie om alerts in te stellen wanneer er bijvoorbeeld 20% meer errors zijn dan de dag ervoor. Het systeem moet filters instellen en iedere minuut checken als basis hebben.	N.V.T.
Applicatie-dashboard	Per applicatie een overzicht (dashboard) om inzichten te krijgen in het aantal fouten, het totale aantal verwerkte jobs, de jobverwerkingstijden, enz. Het dashboard moet inzicht creëren, en het kan een aantal grafieken bevatten, zoals het aantal jobs per uur, fouten per uur en alert notificaties.	/stats, /metrics/jobs, /metrics/queues, /workload
Geschiedenis van jobs	Een functie om de geschiedenis van jobs te kunnen inzien en te kunnen zoeken op ID, gebruiker, datum, naam, enz. Waarschijnlijk een simpele tabel met filter functionaliteiten.	/jobs/{id}
Job-informatie inzien	Een functie om een job te kunnen openen om de precieze stappen in te	/jobs/{id}

	kunnen zien. Hier kunnen bijvoorbeeld constructor data, return data, data die naar de database zou gaan en exception stack traces getoond worden. Dit is de belangrijkste functie van de applicatie.	
Algemeen dashboard	Een algemeen dashboard van alle applicaties met daarin de belangrijkste zaken op één scherm inzichtelijk. Het dashboard kan bijvoorbeeld globale (gegroepeerde) grafieken bevatten en de meest kritische problemen die spelen, zoals ingestelde alerts "+20% more exceptions in the last hour for SomeProject!" of als ergens Horizon niet draait.	N.V.T.

Conclusie

De conclusie is dat Laravel Horizon een breed scala aan functionaliteiten biedt om aan verschillende requirements te voldoen. Uit de opsomming van routes en bijbehorende functies blijkt dat Horizon kan worden gebruikt voor het tonen van dashboards en statistieken per applicatie, het tonen van jobgeschiedenis en het inzien van gedetailleerde job-informatie. Het is ook mogelijk om een algemeen dashboard te maken met globale grafieken en kritieke problemen. Het gebruik van Laravel Horizon voor deze requirements kan leiden tot een effectievere en efficiëntere applicatiebeheer ervaring.

Deelvraag 3

Vraag Welke hiervan kan ik uitbreiden om eventuele requirements te behalen?

Resultaat Zoals je voorheen hebt kunnen zien in de tabel van herbruikbare functionaliteiten zijn er aardig wat zaken die ik vanuit de Laravel Horizon tool kan hergebruiken. Naast hergebruiken wil ik bepaalde functies ook overnemen en vervolgens uitbreiden om andere requirements volledig te behalen.

De functies vanuit de Laravel Horizon package die ik wil uitbreiden en welk doel ik hiermee wil bereiken zal ik hieronder in een tabel visualiseren.

Laravel Horizon Functie?	Wat doet de functie momenteel?	Wat ga ik aanpassen en wat wil ik hiermee bereiken?
RetryController@store	Deze functie retried een gefaalde job.	Ik wil ook pending jobs kunnen retryen dus de functie moet aangepast worden naar dat je ook een status mee kan geven waarop hij checked voor het retrying.
-	-	Een functie maken waarbij er in 1x een hele queue geleegd kan worden.
-	-	Een functie ontwikkelen om alerts aan te maken.
-	-	Een functie ontwikkelen om alerts naar bijvoorbeeld discord te sturen.

Conclusie Er zijn maar een paar functionaliteiten van Laravel Horizon die ik kan hergebruiken of aan kan passen. Verder zijn er nog wel functionaliteiten die ik zelf zal moeten ontwikkelen om alle requirements te behalen.

Conclusie

Algehele conclusie Laravel Horizon biedt een breed scala aan functionaliteiten om aan verschillende requirements te voldoen voor het tonen van dashboards en statistieken per applicatie, het tonen van jobgeschiedenis en het inzien van gedetailleerde job-informatie. Het is ook mogelijk om een algemeen dashboard te maken met globale grafieken en kritieke problemen. Het gebruik van Laravel Horizon voor deze requirements kan leiden tot een effectievere en efficiëntere applicatiebeheer ervaring. Daarnaast is er vastgesteld welke functionaliteiten van Laravel Horizon herbruikbaar zijn en welke mogelijk uitgebreid kunnen worden om eventuele requirements te behalen.

APA Bronvermelding

Laravel - The PHP Framework For Web Artisans. (n.d.). <https://laravel.com/docs/10.x/horizon>

L. (n.d.). *GitHub - laravel/horizon: Dashboard and code-driven configuration for Laravel queues.*

GitHub. <https://github.com/laravel/horizon>