29-10-2018

HOGESCHOOL VAN ARNHEM EN NIJMEGEN

Onderzoeksrapport - Dea

Spring boot

Abdulwahab Alzuabi(591821) | DM.Quentin@student.han.nl

**Versie:** 1.0

**Opdrachtgever:** M. Koolwaaij

[1. Inleiding 2](#_Toc528684290)

[2. Snel van start met Spring Boot 3](#_Toc528684291)

[2.1 Auto-configuration 3](#_Toc528684292)

[2.2 Standalone 3](#_Toc528684293)

[2.3 Opinionated 3](#_Toc528684294)

[3. Configuratie database 6](#_Toc528684295)

[4. Resultaten 7](#_Toc528684296)

[5. Conclusie 8](#_Toc528684297)

# 1. Inleiding

Voor het project “spotitube” was er veel handelingen nodig om de applicatie werkend te krijgen. De stappen om dit te doen was onderander: TomeE downloaden, maven project aanmaken, depencencies toevoegen en voor de run, debug en deployment moesten er instellingen worden geconfigureerd. Dit kost veel tijd en is niet handig qua uren en budget voor de opdrachtgever.

In dit onderzoek wordt er gekeken of spring boot een goede vervangen is voor JEE.

Hoofdvraag:

* Wat zijn de voor en nadelen van spring boot?

Deelvragen:

1. Wat is spring boot?
2. Waarvoor gebruik je spring boot?
3. Wat zijn de voor of nadelen van het configureren van een database met spring boot;

Doelstelling:

Door middel van een framework wordt de startup van een project gemakkelijker en sneller. Zo wordt er tijd en geld bespaard, wat een kan zijn voordeel voor de opdrachtgever.

# 2. Snel van start met Spring Boot

Spring boot is een framework gebouwd op Spring. Het is een eenvoudige manier om snel een applicatie te voorzien van configuraties en voert web gebaseerde applicaties uit.

In JEE moeten deze configuraties zelf geïmplementeerd worden. Dit is een van de grote problemen dat spring boot voor je oplost. Spring boot kiest de dependecies, configureert deze en de applicatie kan gestart worden. Ook maakt Spring Boot het deployment proces gemakkelijker.

Opvallende kenmerken aan spring boot zijn onder andere:

* + - Auto-configuration;
    - Standalone;
    - Opinionated.

Deze kenmerken zal één voor één worden toegelicht.

## 2.1 Auto-configuration

Auto-configuration zorgt ervoor dat de applicatie gebouwd wordt op basis van de gehele omgeving. Dit wordt onder andere gedaan door middel van dependecies die zijn toegevoegd.

Bijvoorbeeld: Het configureren van een database, meer hier over is te vinden in het volgende hoofdstuk (*3. Configuratie database*).

## 2.2 Standalone

Met Spring Boot is het vrij eenvoudig om een stand-alone applicatie te maken. Het proces voor het maken van een Java-based web applicatie is nog al een hele gebeuren.

In het volgende tabel (*Tabel 1 - Handelingen applicatie runnen*) is een vergelijking gelegd tussen JEE en Spring Boot en welke handelingen er nodig zijn om een Java-based applicatie te runnen.

|  |  |  |
| --- | --- | --- |
| Handelingen | JEE | Spring Boot |
| Package de applicatie | X | X |
| Kies webserver en download die | X |  |
| Configureer de webserver | X |  |
| Organiseer de deployment process | X |  |
| Start webserver | X |  |
| Run de applicatie | X | X |

Tabel - Handelingen applicatie runnen

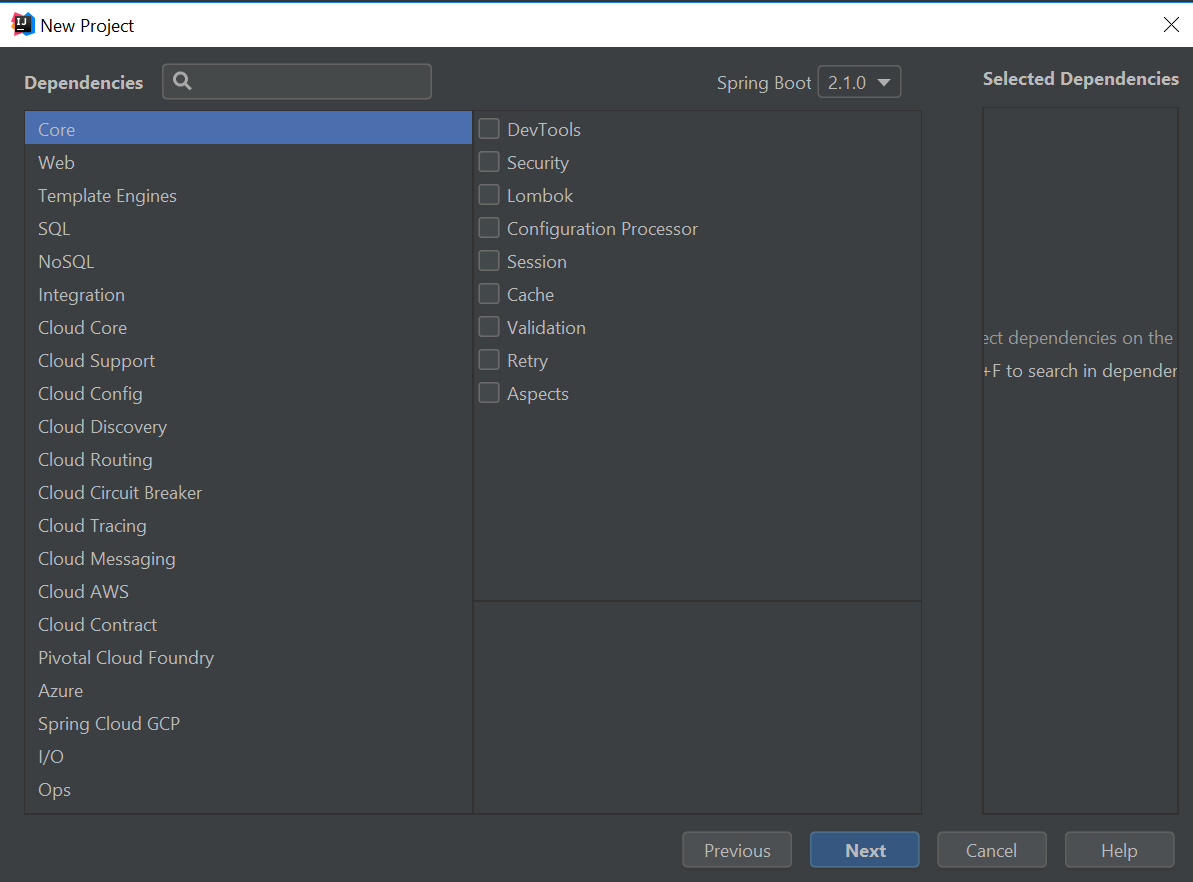
Zoals in het tabel te zien (*Tabel 1 - Handelingen applicatie runnen*) is heeft JEE veel meer handelingen nodig dan Spring Boot. Deze handelingen zijn ook nodig bij Sping Boot alleen worden deze op de achtergrond voor je gedaan.

## 2.3 Opinionated

In de meeste gevallen gebruiken de meeste developers de zelfde populaire libaries. Wat Spring Boot hiermee doet is dat hij ze ophaalt en ze configureert op de meest standaard manier. Hierdoor hoeft de developer niet elke keer bij het starten van een nieuw project de libaries te configureren.

Dit resulteert er in dat de developer meer tijd over heeft voor het schrijven van code.

Een goed voorbeeld hiervan is als je een nieuw Spring Initializr project start met InteliJ, hier kan je alles selecteren wat je nodig hebt en Spring Boot zal het voor je downloaden met de standaard configuratie(*Figuur 1 - Spring Initializr*).



Figuur - Spring Initializr

## 2.4 Spring Framework-technieken

Bij het gebruik van Spring Boot, kan de developer gebruik maken van de standaard Spring Framework-technieken. De “Spotitube” applicatie zou onder andere gebruik van de volgende annotations:

* @Autowired;
* @Component;
* @Controller;
* @Service;
* @Repository;
* @Entity;
* @RequestMapping;
* @Transactional;
* @Modifying;
* @Query.



# 3. Configuratie database

Het configureren van een database is door middel van spring gemakkelijk te behalen. De stappen die hier voor nodig zijn als volgt:

* + - Voeg een dependency toe aan de pom.xml voor de database connector;
    - Maak de database en tabellen zelf(kan ook door middel van entities);
    - Voeg de database properties toe in het bestand “application.properties”.

Dit zijn eigenlijk de enige stappen die nodig zijn om een connectie met een database te leggen. Op de achtergrond zorgt spring boot ervoor dat de database geconfigureerd wordt en klaar is om gebruikt te worden.

In het volgende tabel () is een vergelijking gelegd tussen JEE en Spring Boot en welke handelingen er nodig zijn om een connectie met een database te maken.

|  |  |  |
| --- | --- | --- |
| Handelingen | JEE | Spring Boot |
| Poperties file aanmaken | X |  |
| Database en tabellen aanmaken | X | X |
| Database properties toevoegen | X | X |
| Database gegevens configureren | X |  |
| Connectie starten met database | X |  |
| Connectie sluiten met database | X |  |
| Dependency toevoegen | X | X |

Zoals in het tabel te zien () is heeft JEE veel meer handelingen nodig dan Spring Boot. Deze handelingen zijn ook nodig bij Sping Boot alleen worden deze op de achtergrond al voor je gedaan.

Dit resulteert in minder code, overzichtelijkheid en uitbreidbaarheid voor in de toekomst. Een vergelijking van een stuk code wat precies hetzelfde retourneert is hieronder terug te zien.

JEE:



Spring boot:



# 4. Resultaten

Dit hoofdstuk bevat voorbeelden van het framework Sprint Boot en JEE. Op basis van deze twee voorbeelden wordt er een conclusie vastgesteld. Deze conclusie bevat welke voor en nadelen Sprint boot in vergelijking met JEE heeft.

|  |  |  |
| --- | --- | --- |
| Handelingen | JEE | Spring Boot |
| Principe | 1. Java EE industry approved standard API based framework | 1. Het is gepasseerd op IOC and AOP |
| Injection | 1. @Inject | 1. @Autowired |
| Pattern | 1. JFC MVC framework voor web development | 1. gebruikt Spring DAO framework (gepasseerd op Template design pattern) to connect to database |
| API | 1. JTA API with implementation | 1. Integrates with various Java vendors to support different capabilities such as struts etc |
| License | 1. Oracle license | 1. Open source license |

# 5. Conclusie

Spring boot heeft veel voordelen en het laat de programmeur een makkelijk project configureren en straten.

Zoals te zien is in hoofdstuk (3) is het configureren van een database heel eenvoudig.

Maar wat zijn de voor en nadelen van Spring boot?

Om een tabel te kunnen maken of primary keys te bepalen kan in code geschreven worden op heel eenvoudige manier, de programmeur hoeft alleen maar in de code te schrijven dat hij een tabel of kolom wil maken of een auto increment voor een kolom wil toepassen.

Is werkt applicatie deployment met spring boot?

Spring boot maakt Java applicatie makkelijker te deployment, er hoeft geen configuratie gemaakt te worden, met een klik wordt de applicatie gedeployment.

Sprint boot verminderd deployment, unit test en integratie test tijd doormiddel van default voorzien.

Spring boot verhoogt productiviteit, vervolgens ontwijk het veel code te schrijven en XML te configureren.

Het voorzien veel plug-ins om te developen.

Een belangrijk voordeel van spring boot dat om Spring boot applicatie test is heel makkelijk met gebruik van build tools zoals Maven.

Baeldung: Over spring response header. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://www.baeldung.com/spring-response-header>

Journaldev: Over spring request mapping. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://www.journaldev.com/3358/spring-requestmapping-requestparam-pathvariable-example>

Concretepage: Over spring annotations. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://www.concretepage.com/spring/spring-auto-detection-with-component-service-repository-and-controller-stereotype-annotation-example-using-componentscan-and-component-scan>

Dzone: Over spring boot voordelen. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://dzone.com/articles/why-springboot>

Springframework: Over spring boot annotations. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://springframework.guru/spring-framework-annotations/>

Spring: Over spring boot auto configuration. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://docs.spring.io/spring-boot/docs/current/reference/html/using-boot-auto-configuration.html>

Spring: Over spring boot dependency. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://docs.spring.io/spring-boot/docs/current/reference/html/using-boot-spring-beans-and-dependency-injection.html>

Github: Over spring boot en JavaEE. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://ssricardo.github.io/2017/javaee-spring-boot/>

Dzone: Over spring boot uitleggen. (z.d.). Geraadpleegd op 30 Oktober 2018, van <https://dzone.com/articles/what-is-spring-boot>