Spring Boot, REST & Postman opdracht

**Je kan een Spring Boot applicatie packagen naar een JAR of een WAR. Kun je uitleggen wat deze package types inhouden en wat "officieel" het verschil is?**

Een springboot applicatie kan gepackaged worden in een JAR en daar zitten alle modules/service’s dependencies meteen in. Deze kan dan overal gerund worden met java -jar <jar-path>. Een WAR daarin tegen is meer voor web containers. Dit bevat niet alle dependencies erin maar die worden opgehaald door bijvoorbeeld een tomcat.

**Wat doet de @Controller annotatie van Spring Boot? En de @RestController annotatie?**

De @Controller annotatie zet een klasse om in een controller klasse. Deze wordt automatisch gedetecteerd tijdens classpath scanning. De @RestController annotatie zet een klasse om in een restful controller.

**Leg in je eigen woorden uit wat de volgende HTTP methodes inhouden:**

**GET:** Dit haalt informatie op van een request en hoort nooit iets aan te passen waar je naartoe request.

**POST:** Dit stuurt informatie mee met een request waar vervolgens de back-end iets mee kan doen. Bij het invoegen van iets.

**PUT:** Als je van een bestaand object bijvoorbeeld een aantal waarden wilt veranderen gebruik je het PUT request.

**DELETE:** Als je data in de back-end wilt verwijderen door middel van een request gebruik je de delete methode.

**PATCH:** Patch lijkt erg op PUT maar het kan ook effecten hebben op andere resources.

**Leg kort uit waar de volgende HTTP response codes voor dienen:**

**200:** Dit geeft aan dat de request goed is verlopen en alles terug is gestuurd.

**301:** Geeft aan dat de informatie van die url is verplaatst naar een andere url en niet meer beschrikbaar is op de huidige.

**401:** Geeft aan dat de gebruiker geen toegang heeft om deze request te doen. De gebruiker is niet geautoriseerd of gemachtigd.

**403:** Dit betekend dat je geen toegang hebt tot de opgegeven URL

**418:** De klassieke “I’m a teapot” error. Volgensmij was dit als een grap bedoeld. Weet niet of het echt gebruikt word.

**500:** Geeft aan dat er een onbekende fout is opgetreden en dat je er niet uit op kan halen wat de fout is.

**Vaak worden requests in een log opgeslagen. (Voorbeeld:** [**https://github.com/elastic/beats/blob/master/filebeat/module/nginx/access/test/access.log (Koppelingen naar een externe site.)**](https://github.com/elastic/beats/blob/master/filebeat/module/nginx/access/test/access.log)**). Stel je maakt een REST endpoint voor het inloggen van een gebruiker, zou je met inachtname van security dan voor een GET of een POST methode kiezen en waarom?**

Met een post request stop je de informatie die je wilt niet in een URL zoals bij een GET request. Ook kan je een GET request gewoon doen in the URL bar van de meeste browsers waar je een POST echt moet versturen via de logica.

**Wat betekent RESTful en wat houd dat in?**

Een RESTful service is een service die een pattern implementeerd van REST. Dit houd in dat de architectuur van deze twee verschillend is.