

# SYT GK771 Windpark REST

## Alan Matykiewicz

Man verwendet die `SY4_WINDPARK_DEMO_REST_JSON.zip` Spring-Boot Applikation als Startpunkt. IntelliJ kann solche Applikationen selbständig starten. Alternativ kann die auch mittels `mvn spring-boot:run` laufen lassen.

Im Startpunkt ist der Generator zur Erzeugung der Winddaten bereits implementiert. Das Wichtige befindet sich in der WindengineController Klasse. Die Klasse `windengineDataXML` erstellt eine Seite wo die Winddaten im xml Format angezeigt werden. Wichtig hier ist die **RequestMapping** Zeile.

```
@RequestMapping(value="/windengine/{windengineID}/xml", produces =  
{"application/xml"})
```

Value bestimmt den Pfad. Produces definiert, dass hier die Daten im `xml` Format angezeigt werden. Da die WindengineData Klasse die `@XmlRootElement` Markierung hat, ist dieses Daten formatieren möglich. Ähnlich funktioniert das für Daten im `json` Format innerhalb der `windengineDataJSON` Klasse.

```
@RequestMapping(value="/windengine/{windengineID}/json", produces =  
{"application/json"})
```

Zuletzt müssen bei der **MainPage** die Hyperlinks angepasst werden.

```
String mainPage = "This is the windengine application! (DEZSYS_GK72_WINDPARK)  
<br/><br/>" +  
                "<a  
href='http://localhost:8080/windengine/001/xml'>Link to windengine/001/xml</a>  
<br/>" +  
                "<a  
href='http://localhost:8080/windengine/001/json'>Link to windengine/001/json</a>  
<br/>";
```