Swagger:

[Swagger](https://github.com/wordnik/swagger-core) does a simple but very powerful thing: with a bit of additional annotations it generates the [REST](http://en.wikipedia.org/wiki/Representational_state_transfer) API descriptions ([HTTP](http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) methods, path / query / form parameters, responses, [HTTP](http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) error codes, ...) and even provides a simple [web UI](https://github.com/wordnik/swagger-ui) to play with [REST](http://en.wikipedia.org/wiki/Representational_state_transfer) calls to your APIs (not to mention that all this metadata is available over [REST](http://en.wikipedia.org/wiki/Representational_state_transfer) as well).

Swagger console is used for:

* description of the service
* relative context path
* parameters (form / path / query), required or optional
* [HTTP](http://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) status codes: **201 CREATED** and **409 CONFLICT**
* ready to go **Try it out!** to call [REST](http://en.wikipedia.org/wiki/Representational_state_transfer) service immediately (with out-of-the box parameters validation)

Implementation:

 Firstly, every single [JAX-RS](https://jax-rs-spec.java.net/) service which supposed to be documented should be annotated with **@Api**annotation