

## Article

[Pablo Frigolett](#) · Mar 7, 2023 3m read

## spec for swagger-ui from spec class

Surely you wanted to use the OpenAPI Specification (OAS) JSON you used for your spec class on the iris-web-swagger-ui iris package. The generated OAS from the `##class(%REST.API).GetWebRESTApplication(...)` method is very crude, with no description on the parameters or the expected response structure.

So after creating your REST application from an OAS you should have:

- A generated application.disp.cls class
- An application.spec.cls class (we'll add a path to the OpenAPI spec that's residing in the "XData OpenAPI" property)
- An application.impl.cls class (we'll just implement the GetSpec method)

If you installed the iris-web-swagger-ui package

(<https://openexchange.intersystems.com/package/iris-web-swagger-ui>) you need an endpoint that returns the OAS

My approach is to add a path to the OAS in the spec class and implement it using the property "OpenAPI" from said spec class. This will get you, for instance, every structure in the original OAS available for testing!

In the sample class for the iris-web-swagger-ui package (Sample.PersonREST), there's an implementation for the SwaggerSpec method that updates some properties to reflect what's configured in the IRIS Web Application. If you're following this post, I recommend updating directly the OAS in the spec class to provide that information in order to centralize your API's documentation (or you can always update it in the GetSpec method I provide as sample).

Following the example in the Sample.PersonREST class, the path will be `"/spec"`. Although there is a value for the "basePath" tag, it will be overridden with the IRIS Web Application name:

```
Class application.spec Extends %REST.Spec [ProcedureBlock]
{

XData OpenAPI [ MimeType = application/json ]
{
{
...
"basePath": "/api/myapi",
...
"paths": {
"/_spec": {
"get": {
"operationId": "GetSpec",
"description": "This OpenAPI Spec",
"produces": [
"application/json"
],
"responses": {
"200": {
"description": "Successful operation"

```

• • •

```

''' This OpenAPI Spec
ClassMethod GetSpec() As %DynamicObject
{
    Set spec = {}.%FromJSON(##class(%Dictionary.CompiledXData).%OpenId("application.s
pec|OpenAPI").Data.Read())
    Set spec.basePath = %request.Application
    Return spec
}

```

Now go to your iris-web-swagger-ui (in my case <http://localhost:52773/swagger-ui/index.html>) and test away exploring at "<http://localhost:52773/api/myapi/spec>" in swagger-ui's navigation input!

## #REST API #InterSystems IRIS

Source URL:<https://community.intersystems.com/post/spec-swagger-ui-spec-class>