

WEB DAY

2022

ONLINE CONFERENCE



AZURE FUNCTIONS E OPENAPI : LA COPPIA PERFETTA!!

MASSIMO BONANNI

TECHNICAL TRAINER
MICROSOFT

@MASSIMOBONANNI

10 MARZO 2022


#WEBDAY2022

Kudos to Sponsors

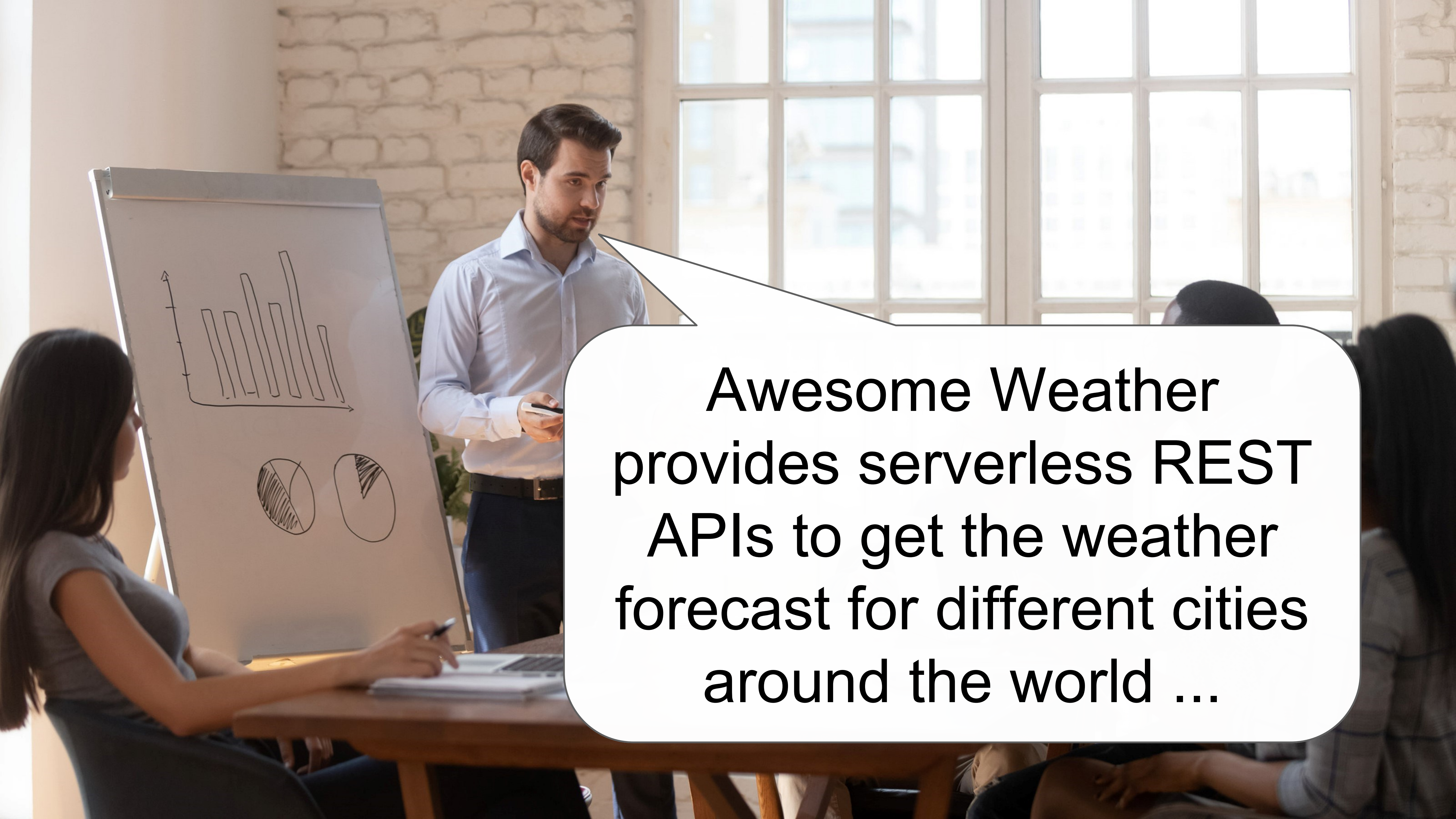


CODICEPLASTICO

managed/designs

A blurred background image of a meeting room. Several people are seated around a wooden table. On the left, a woman is looking at a whiteboard that has a line graph on it. Large windows in the background let in bright light. The text is overlaid on a white rounded rectangle in the center.

Someday ...
... in one of Awesome
Weather's meeting
rooms ...

A man in a light blue shirt is standing and presenting to a group of people in a meeting room. He is holding a small object in his hand. To his left is a whiteboard with a bar chart and two pie charts. A woman is sitting at a table in the foreground, looking at the whiteboard. The room has large windows in the background.

**Awesome Weather
provides serverless REST
APIs to get the weather
forecast for different cities
around the world ...**



... now we want to deliver the APIs as SaaS and we need to document them to let our clients know how to use them ...



Any idea on how
to do it?


Doc or pdf documents?

No, doc or pdf is not
the right solution

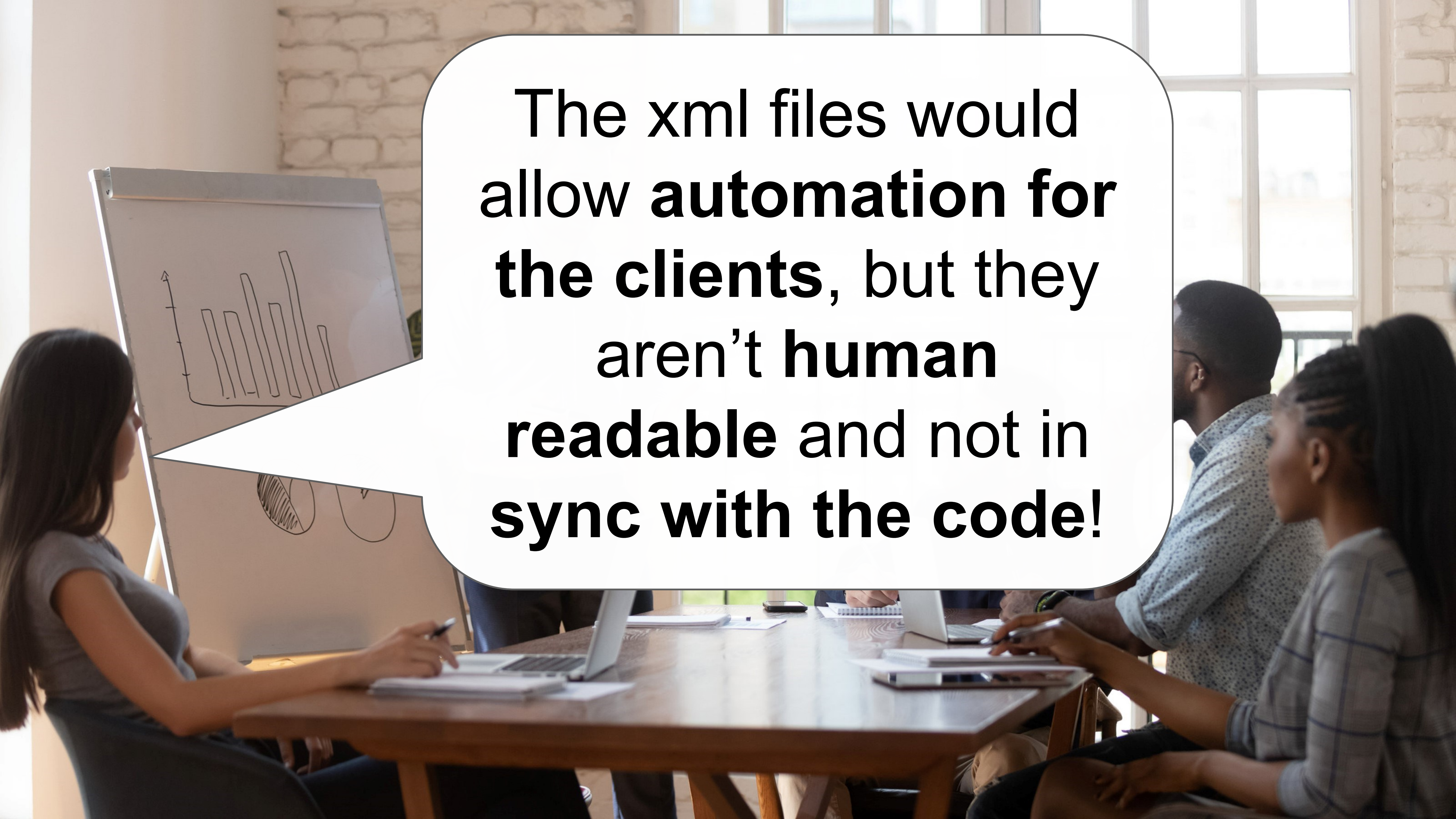


... they are not **sync**
with code!
They don't allow
automation for clients!





Can we
produce xml
files?

A group of people are seated around a wooden conference table in a bright room with large windows. A woman on the left is looking at a whiteboard that displays a bar chart and a diagram. A large, white, rounded rectangular text box is superimposed over the center of the image, containing the text:

The xml files would allow **automation for the clients**, but they aren't **human readable** and not in **sync with the code!**

The xml files would allow **automation for the clients**, but they aren't **human readable** and not in **sync with the code!**

A group of people are gathered around a wooden conference table in a bright, modern office. A woman on the left is looking at a whiteboard that displays a bar chart and two pie charts. A man in the center is looking at a laptop. A woman on the right is looking at a laptop. A large, white, rounded rectangular overlay is positioned in the center of the image, containing the text "OpenAPI is the answer!!!".

**OpenAPI is
the answer!!!**

What is the OpenAPI Specification?

#WEBDAY2022

The **OpenAPI Specification** (OAS) defines a standard, programming language-agnostic interface description for HTTP APIs.



Advantages of OpenAPI specification

#WEBDAY2022

1

Allows you to interact with the remote service without understand implementation logic

2

Removes guesswork in calling a service

3

Allows you to generate SDKs automatically from the definition

OpenAPI terminology

#WEBDAY2022

OpenAPI specification

It is the standard for describing a RESTful API.
It can be seen as your guide for writing the API definition following the OpenAPI rules.
AKA OpenAPI specs.

OpenAPI definition

It describes exactly how your API works.
It is written following the OpenAPI specification.
AKA OpenAPI contract.

OpenAPI documentation

It is human-readable documentation for your API based on your OpenAPI definition.
It can be automatically generated from your OpenAPI definition to avoid the pain of writing it by hand.
AKA OpenAPI reference.

How configure Azure Functions to provide OpenAPI

#WEBDAY2022

In-Process

`Microsoft.Azure.WebJobs.Extensions.OpenApi`
It supports only Azure Functions v2 and later

Isolated

`Microsoft.Azure.Functions.Worker.Extensions.OpenApi`
It supports only Azure Functions v3 and later

OpenAPI endpoints

Provides the OpenAPI (version 2.0 or 3.0) definition of your functions. You can choose YAML or JSON.

```
openapi: 3.0.1
info:
  title: Awesome Weather REST API
  description: Awesome Weather provides the best API to get updated and accurate weather forecasts. the
  termsOfService: https://github.com/massimobonanni/AzureFunctionOpenAPI
  contact:
    name: AwesomeWeather
    url: https://github.com/massimobonanni/AzureFunctionOpenAPI
    email: info@awesomeweather.com
  license:
    name: MIT
    url: http://opensource.org/licenses/MIT
  version: 1.0.0
servers:
  - url: http://localhost:7071/api
paths:
  /weatherforecasts:
    get:
      tags:
        - WeatherForecast
      summary: Returns the weather forecasts of cities
      description: Returns the weather forecast for the next 5 days for cities that have the filter parameter in their name. If the
      filter is not set, it returns the weather forecast for all cities.
      operationId: getcityforecasts
      parameters:
        - name: cityName
          in: query
          description: If this parameter is used, the weather forecast for cities that have the parameter content in the name will be
          returned
      schema:
        type: string
        x-ms-summary: Possible filter for the name of the city
        x-ms-visibility: important
      responses:
        '200':
          description: List of cities resulting from the search and for each city the forecast for the next 5 days
```

Provides the OpenAPI (version 2.0) definition of your functions. You can choose YAML or JSON.

```
{
  "swagger": "2.0",
  "info": {
    "title": "Awesome Weather REST API",
    "description": "Awesome Weather provides the best API to get updated and accurate weather
forecasts. the",
    "termsOfService": "https://github.com/massimobonanni/AzureFunctionOpenAPI",
    "contact": {
      "name": "AwesomeWeather",
      "url": "https://github.com/massimobonanni/AzureFunctionOpenAPI",
      "email": "info@awesomeweather.com"
    },
    "license": {
      "name": "MIT",
      "url": "http://opensource.org/licenses/MIT"
    },
    "version": "1.0.0"
  },
  "host": "localhost:7071",
  "basePath": "/api",
```

Function Runtime Version: 4.0.1.16815

Functions:

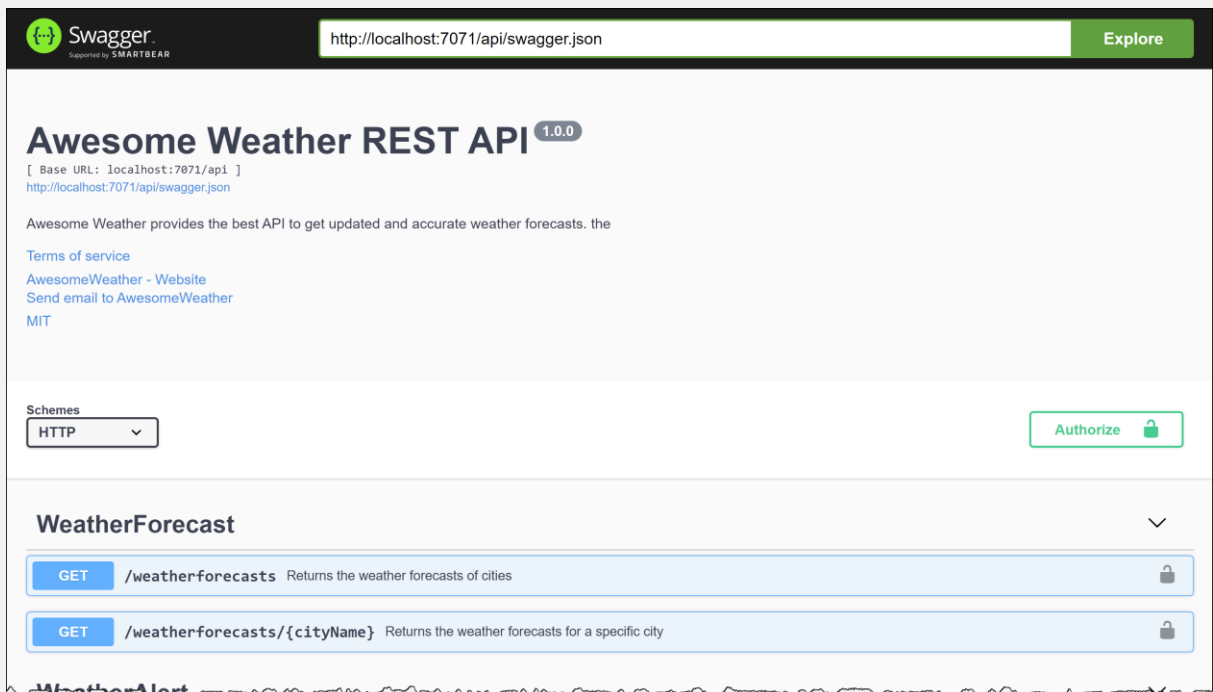
CreateCityForecastAlert: [POST] http://localhost:7071/api/weatherforecasts/alerts
GetCityForecast: [GET] http://localhost:7071/api/weatherforecasts/{cityName}
GetCityForecasts: [GET] http://localhost:7071/api/weatherforecasts
RenderOAuth2Redirect: [GET] http://localhost:7071/api/oauth2-redirect.html

RenderOpenApiDocument: [GET] http://localhost:7071/api/openapi/{version}.{extension}

RenderSwaggerDocument: [GET] http://localhost:7071/api/swagger.{extension}

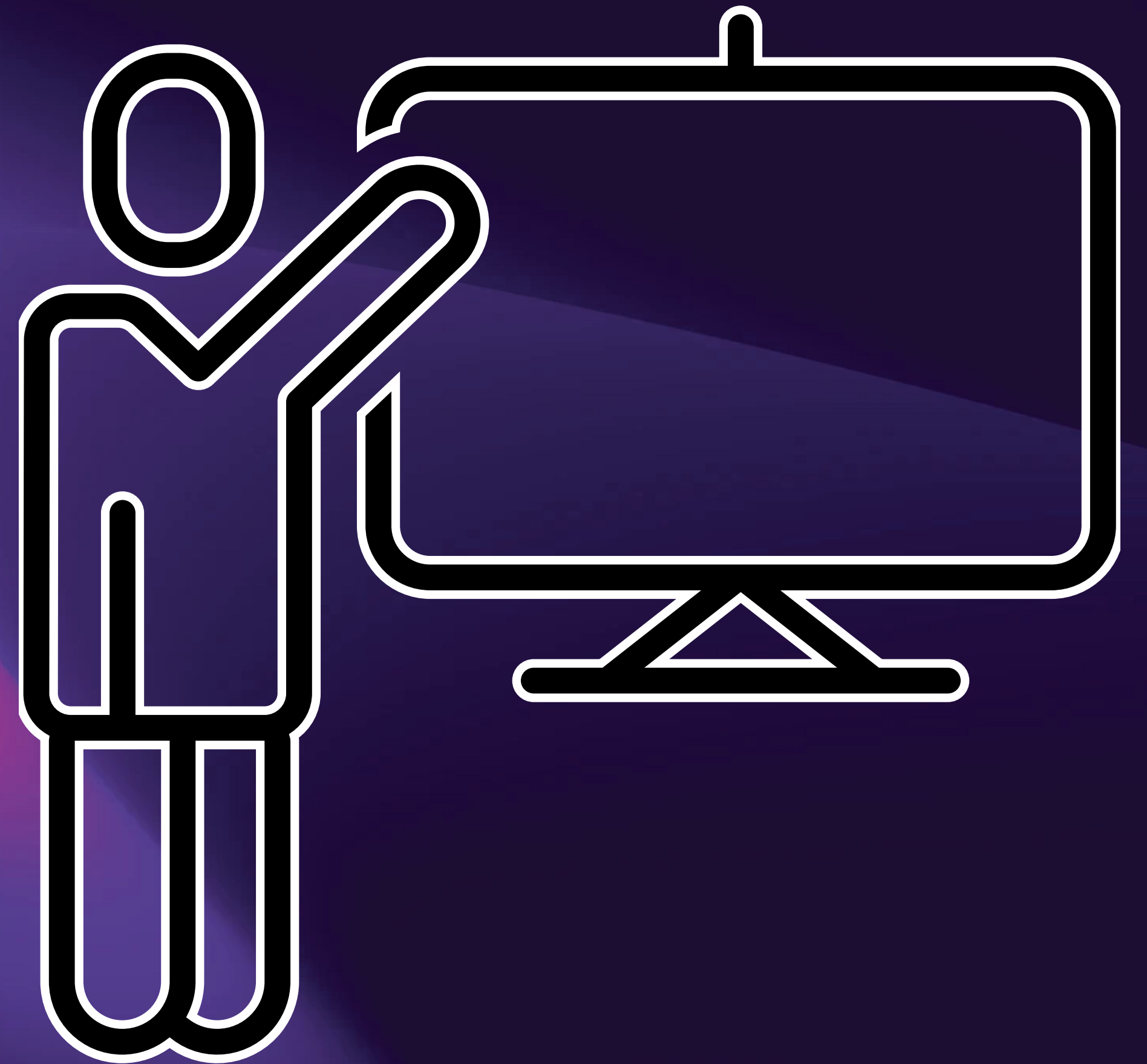
RenderSwaggerUI: [GET] http://localhost:7071/api/swagger/ui

Provides the OpenAPI documentation of your functions.



DEMO

FROM ZERO TO FULL OPENAPI SUPPORT IN AZURE FUNCTIONS!




```
{  
  "IsEncrypted": false,  
  "Values": {  
    "OpenApi__ApiKey": "123456789",  
    "OpenApi__AuthLevel__Document": "Function",  
    "OpenApi__AuthLevel__UI": "Function"  
  }  
}
```

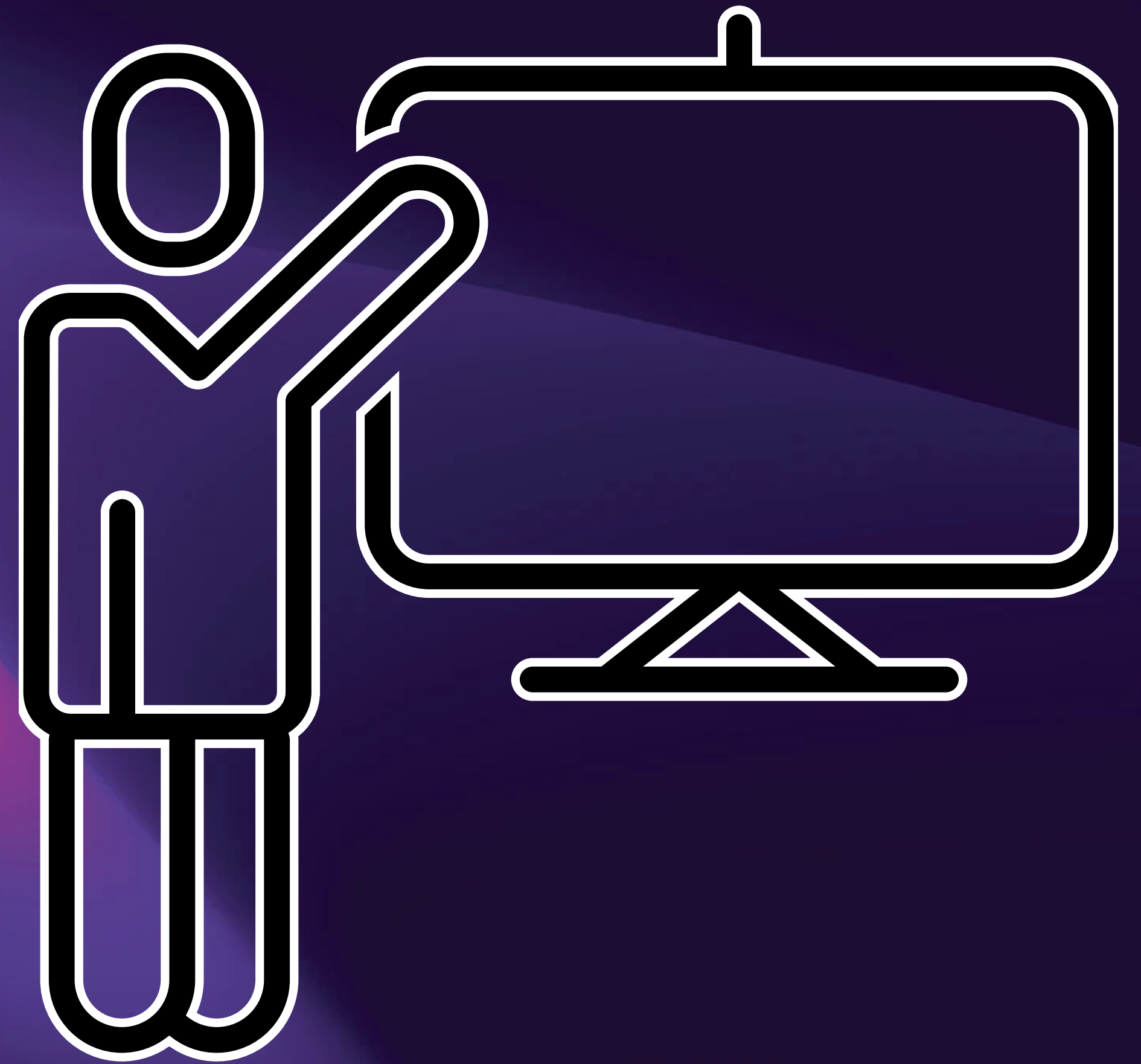
You can configure API Key (like your Azure Functions) defining it in the appsettings.json

You can implement your own authentication process overriding the default authorization.

```
public class OpenApiHttpTriggerAuthorization : DefaultOpenApiHttpTriggerAuthorization  
{  
    public override Task<OpenApiAuthorizationResult> AuthorizeAsync(IHttpRequest request)  
    {  
        var result = default(OpenApiAuthorizationResult);  
        // ...  
        return Task.FromResult(result);  
    }  
}
```


DEMO

**OPENAPI ENDPOINTS SECURITY IN
AZURE FUNCTIONS!**



Conclusions

#WEBDAY2022

OpenAPI is an open specification backed by big companies, and, over time, the implementation is stable

OpenAPI specification is language-agnostic, and does not require rewriting existing APIs

Language-agnostic is great for inclusivity, but can introduce a complexity when you use it in language-specific frameworks

Take care of the learning curve. For developers unfamiliar with its concepts, OpenAPI can function as a bandaid rather than a toolset

Thanks for your attention!!!!!!



Massimo Bonanni

Microsoft Technical Trainer

massimo.bonanni@microsoft.com

@massimobonanni

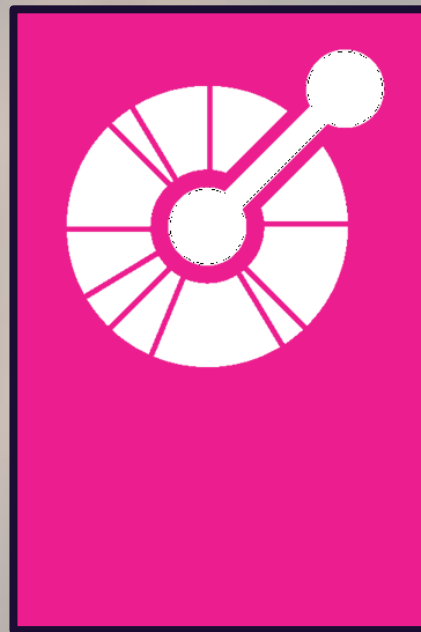


WEB DAY
2022

Foto di Pete Linforth da Pixabay

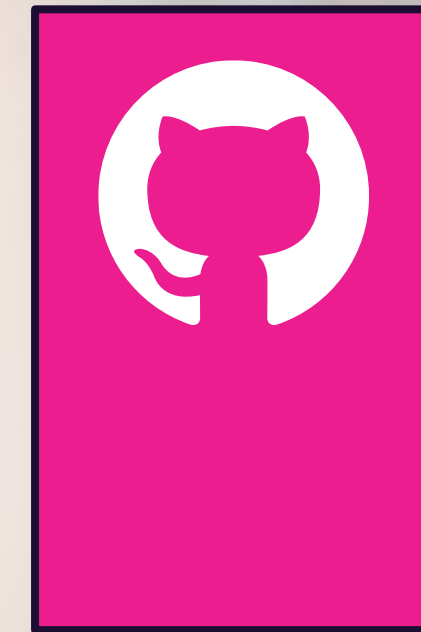
References

#WEBDAY2022



OpenAPI Initiative

<https://www.openapis.org/>



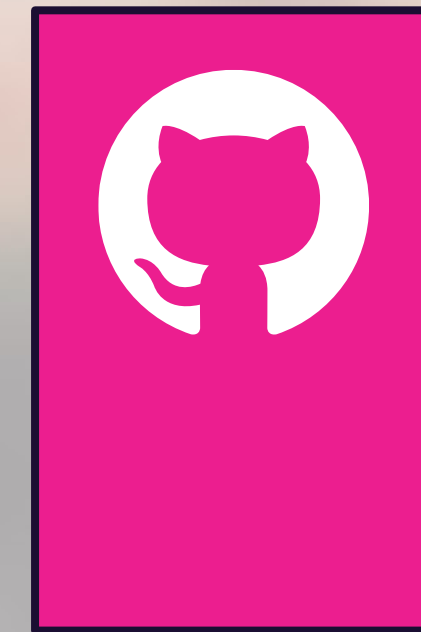
OpenAPI Specification – GitHub Repo

<https://github.com/OAI/OpenAPI-Specification/>



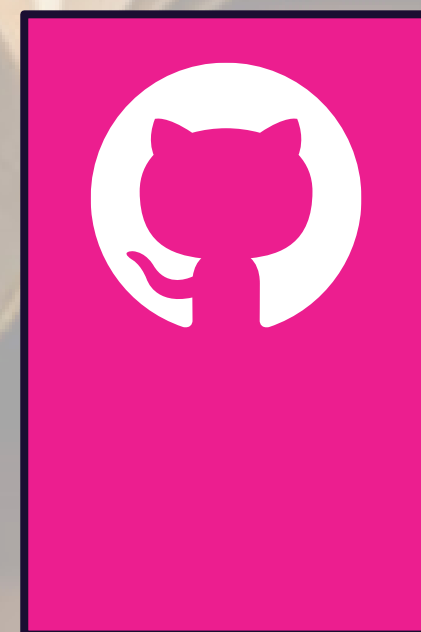
Deploy OpenAPI enabled Azure Functions with .NET in Visual Studio

<https://docs.microsoft.com/en-us/shows/Azure-Friday/Deploy-OpenAPI-enabled-Azure-Functions-with-NET-in-Visual-Studio>



Azure Functions OpenAPI Extension – GitHub Repo

<https://github.com/azure/azure-functions-openapi-extension>



Demo- GitHub Repo

<https://github.com/massimobonanni/AzureFunctionOpenAPI>