



# ASP .NET Core REST API tips, tricks & tools

---

Jakub Wajs

# Agenda

- Intro
- Project structure
- Data Transfer Objects (DTO)
- Caching, paging, logging
- API Versioning
- Automatic documentation
- Outro + Q&A



# Jakub Wajs

.NET Developer @ Emakina.PL

---

LinkedIn: [www.linkedin.com/in/jakub-wajs](https://www.linkedin.com/in/jakub-wajs)

Blog: [jwajs.net](https://jwajs.net)

GitHub: [github.com/kubawajs](https://github.com/kubawajs)

---

# Github URL

[github.com/kubawajs/PutNetPresentation](https://github.com/kubawajs/PutNetPresentation)

# Project structure

***Separation of Concerns - design principle for separating a computer program into distinct sections such that each section addresses a separate concern.***

---

[Wikipedia, Separation of concerns](#)

---

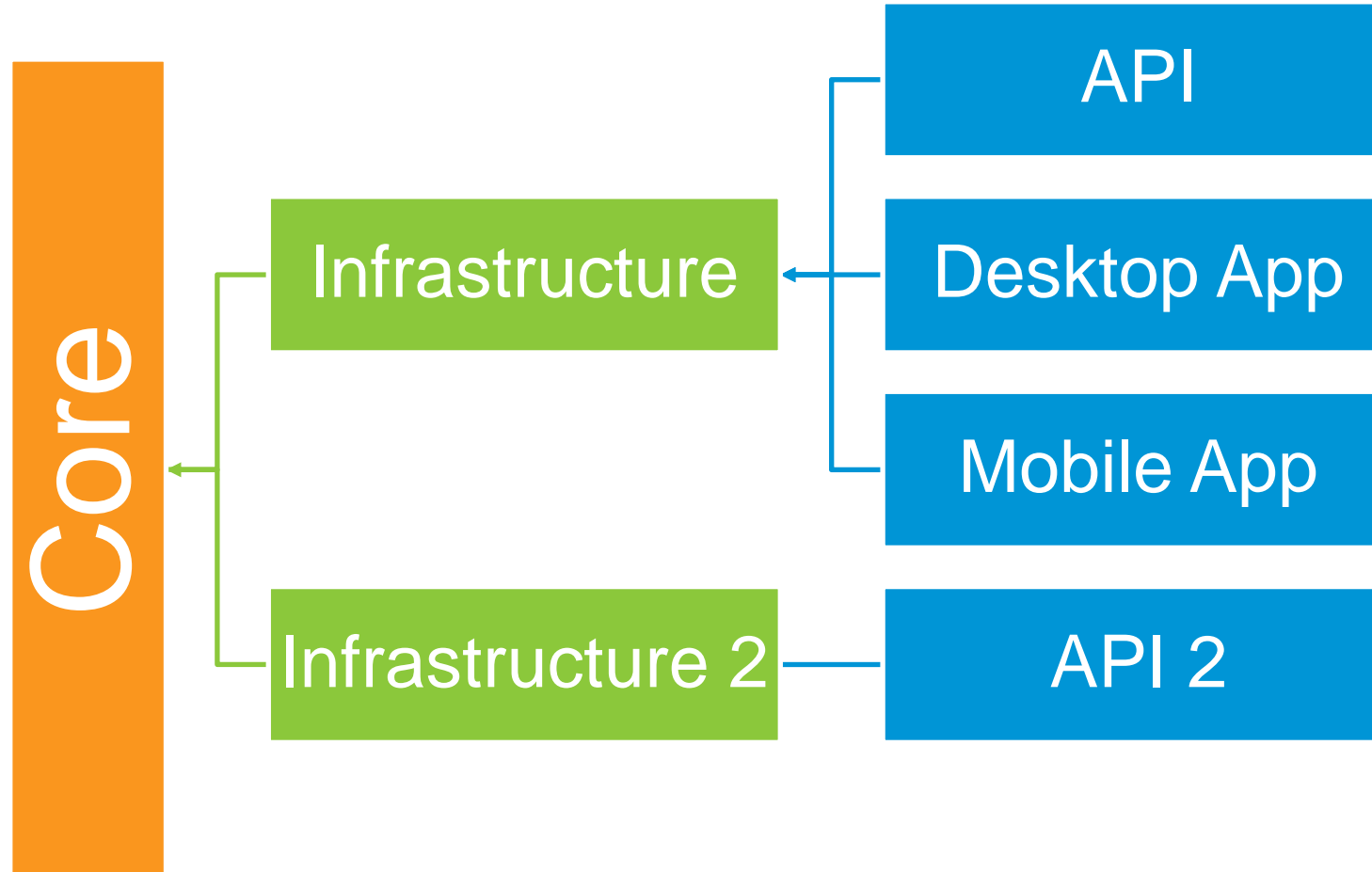
# Project structure

Dividing: entities, business logic, database access

Reusability of components

Ease of understanding

# Project reference order





# Demo

Project structure

# Data Transfer Objects

---

Models, DTOs & Automapper

# What's wrong with models?

The representation of an object in the database table may contain unnecessary (or forbidden!) data

POST and PUT requests with model class as parameter can lead to uncontrolled changes in the database

Change in database affects model class

A lot of data to send!

*A Data Transfer Object is an object that is used to encapsulate data and send it from one subsystem of an application to another.*

---

[Benny Hallett @ stackoverflow.com](#)

# Data Transfer Objects

The representation of an object in the database table may contain unnecessary (or forbidden!) data

POST and PUT requests with model class as parameter can lead to uncontrolled changes in the database

Adding new column in database affects model class

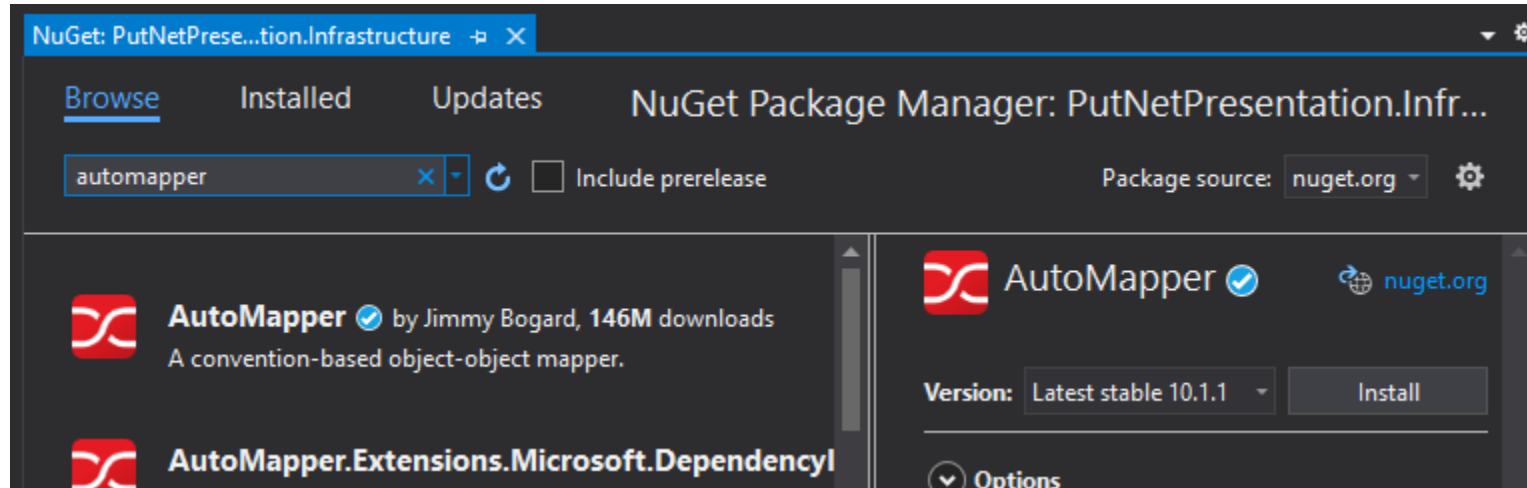
A lot of data to send!

Separate model for GET, POST and PUT requests

Adding new column does not affect DTO

Only as much information as required is send

# Automapper



- NuGet Package Manager: **AutoMapper & AutoMapper.Extensions.Microsoft.DependencyInjection**
- Documentation: <https://automapper.org/>
- Github: <https://github.com/AutoMapper/AutoMapper>

# Demo

---

AutoMapper & .NET 5

# Caching, paging, logging





of visits are **abandoned** if a mobile site takes longer than **3 seconds to load**.

### Think with Google

Google Data, Global, n=3,700 aggregated, anonymized Google Analytics data from a sample of mWeb sites opted into sharing benchmark data, March 2016.

# Caching

Performance improve by reducing time and resources required to generate content

Works best with data that changes infrequently and is hard to generate

# Caching types

- Response caching
- In-memory caching
- Distributed caching

# Paging

**Definition:**

*Pagination is an ordinal numbering of pages, which is usually located at the top or bottom of the site pages.*

[sitechecker.pro/pagination](https://sitechecker.pro/pagination)



<https://localhost:44353/api/users?pageNumber=2&pageSize=25>

# Logging

Fundamental for application monitoring

Allows to investigate errors that happened in the past

Sometimes only way for debugging (Production environments)

# Nlog



**NLog** by Jarek Kowalski, Kim Christensen, Julian Verdurmen

v4.7.9 ✖

NLog is a logging platform for .NET with rich log routing and management capabilities. NLog supports traditional logging, structured logging and the combination of both.



**NLog.Web.AspNetCore** by Julian Verdurmen

v4.12.0

NLog LoggerProvider for Microsoft.Extensions.Logging and ASP.NET Core platform. Adds helpers and layout renderers for websites and web applications.

- NuGet Package Manager: **NLog & Nlog.Web.AspNetCore**
- Documentation: <https://nlog-project.org/>
- Step by step setup tutorial for .NET 5: [Getting started with ASP.NET Core 5](#)

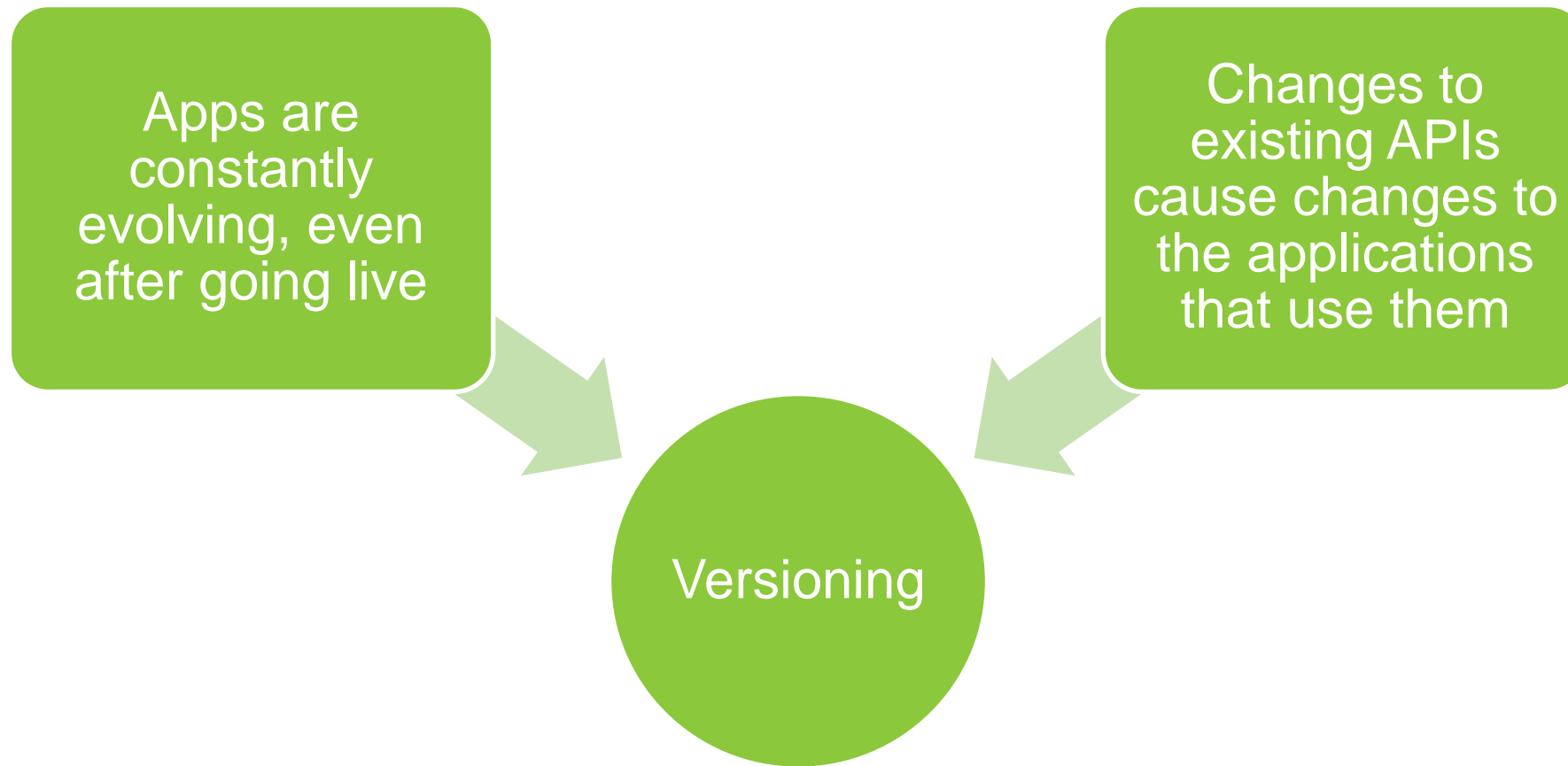
# Demo

Caching, paging, logging

# API Versioning

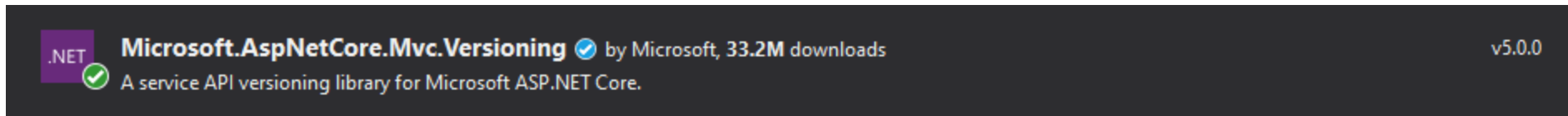


# API Versioning



# Versioning in .NET 5

- NuGet Package Manager: **Microsoft.AspNetCore.Mvc.Versioning**



# Demo

---

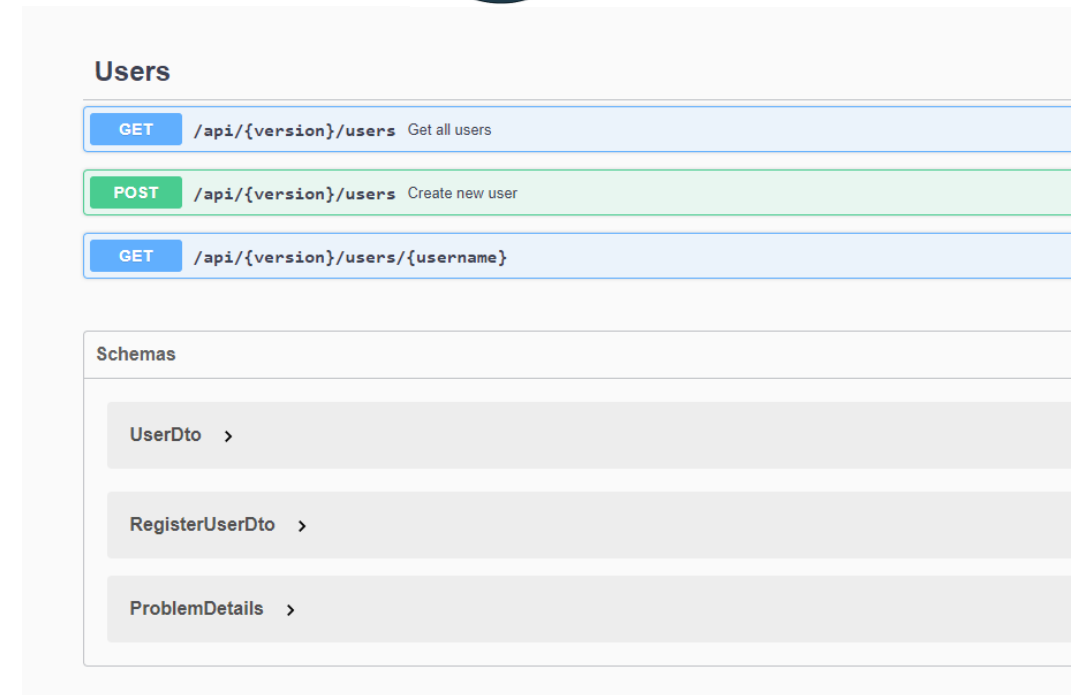
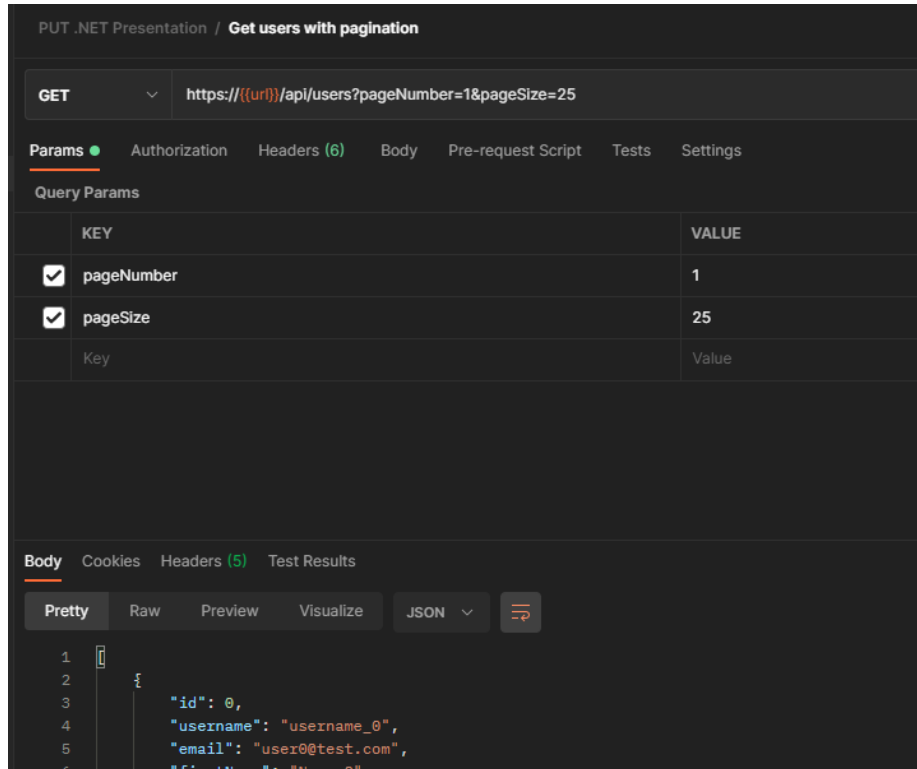
API Versioning in .NET 5

# Automatic documentation

---

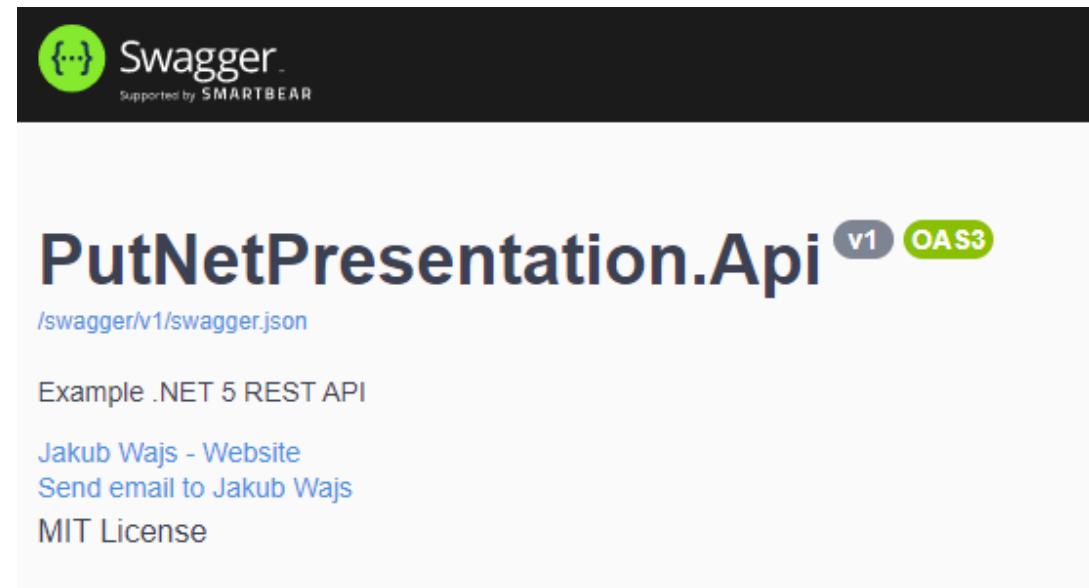
Swagger & Postman

# Automatic documentation



# Adding OpenApiInfo

```
services.AddSwaggerGen(c =>
{
    c.SwaggerDoc("v1", new OpenApiInfo
    {
        Title = "PutNetPresentation.Api",
        Version = "v1",
        Description = "Example .NET 5 REST API",
        Contact = new OpenApiContact
        {
            Name = "Jakub Wajs",
            Email = "j.wajs@emakina.pl",
            Url = new Uri("https://www.linkedin.com/in/jakub-wajs/")
        },
        License = new OpenApiLicense
        {
            Name = "MIT License"
        }
    });
});
```



# XML comments

```
/// <summary>
/// Create new user
/// </summary>
/// <param name="registerUserDto"></param>
/// <remarks>
/// Sample request:
///
///     POST /users
///     {
///         "username": "user_123",
///         "email": "user@mail.com",
///         "password": "Ex4mplePas$"
///     }
///
/// </remarks>
/// <returns></returns>
/// <response code="201">Returns the newly created item</response>
/// <response code="400">If the item model is invalid</response>
[HttpPost]
[ProducesResponseType(typeof(StatusCodes.Status201Created))]
[ProducesResponseType(typeof(StatusCodes.Status400BadRequest))]
0 references
public async Task<IActionResult> Register([FromBody]
```

**POST** /api/users Create new user

Sample request:

```
POST /users
{
  "username": "user_123",
  "email": "user@mail.com",
  "password": "Ex4mplePas$"
}
```

# Demo

---

Swagger & Postman



# Outro

---

# Summary

Maintaining an appropriate project structure is important from the very beginning

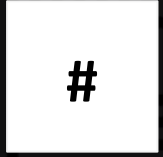
Never use domain model classes, use DTOs!

Caching and paging improves performance

Without logging your API is a black box

Versioning allows you to extend the API without affecting the users already using it

Without documentation even the best API is worth nothing



# Q&A

---

# Thank you and stay in touch!



Jakub Wajs

.NET Developer @ Emakina.PL

[www.linkedin.com/in/jakub-wajs](https://www.linkedin.com/in/jakub-wajs)

[jwajs.net](https://jwajs.net)

[github.com/kubawajs](https://github.com/kubawajs)

---