Anton Martyniuk • Following
Microsoft MVP | Helping 30K+ Software Engineers Improve .NET Skills and Craft Better Sof...
View my newsletter
51m •

Why I Don't Use Data Annotations In My Projects

The best way to validate objects in 2025 👇

Data Annotations let you enforce validation rules with attributes on your models.

While DataAnnotations might seem like a concise way to define validation inside your models, this approach has serious drawbacks:

- Tight Coupling: validation lives inside your entities, so swapping frameworks means touching every model.

- Mode Sharing: often public models are shared across different projects as project or Nuget references. Dependencies on validation attributes and specific versions of Microsoft packages can make assembly sharing difficult.

- Limited Flexibility: Data annotations provide limited flexibility compared to FluentValidation configurations. Complex rules are hard, sometimes impossible, to express with attributes alone.

- Code Clutter: mixing validation with models violates Single Responsibility and makes the code harder to maintain.

- Limited Reusability: cross-property checks or shared rules across entities become almost impossible to implement.

As requirements grow, Data Annotations turn from small and handy to headache, especially when one rule depends on many fields.

That's why I use **FluentValidation** library in the production.

With FluentValidation you keep models clean and write expressive rules in separate classes:

FluentValidation offers the following advantages:

- Flexibility: chain conditions, custom logic, and async checks with ease.

- Separation of Concerns: validation sits outside domain models, making code more readability and more testable.

- Explicitness: rules read like plain English; new devs "get it" at first glance

- Reuse and Composition: share validators, compose child rules, and keep large projects sane.

Do you also use FluentValidation in your projects? Leave down the comment 👇

✅ If you like this post - **repost** to your network and **follow** me

✅ 10 seconds to join **6**,**000**+ readers in my free newsletter to improve your .NET skills and learn how to craft better software:

https://lnkd.in/e3nbKff5

60 39 comments 3 reposts