

[Skip to main content](#)

r/golang



Search in r/golang

[Log In](#)

r/golang • 20 days ago

NiceCelery499



## Go + htmx + templ library is awesome, anyone else has test this way?

Can response the htmx endpoints with render the tpl directly, is very good really, anyone who has used leave here your thoughts about it

134

75

Share

Sort by: **Best** ▾[+](#) Add a Comment

gnick666 • 20 days ago

I'm not using templ, just the stdlib templates, apart from that, it's a blast to work with 😄

Obligatory project link: <https://github.com/blackfyre/wga>

15

Reply

Share



4 more replies



grahaman27 • 20 days ago

JavaScript is annoying but not a deal breaker. The issue for me is the cruft and complexity of frameworks.

I'm debating whether to try using this, I want a simpler way to build websites. But I'm worried the templ and htmx markup language is going to be more annoying than JavaScript.

Can someone help clarify the main benefit here? Will this make it simpler or just a new standard I will have to learn over and over again?



25

Reply

Share



kaeshiwaza • 20 days ago

The logic of HTMX and server side rendering with templates from backend is nothing new, it's very old since the begin of html. We was using (i)frame to update small parts of a page, then ajax and then htmx just to make this easier, nothing more. Part of htmx will probably be integrated in html specs one time because it doesn't change the logic, like was part of jquery.

20

Reply

Share



3 more replies



Swoo413 • 20 days ago

You can just use htmx with the std library templating package instead of using templ. That's what I've been doing and it works great. Htmx for reactivity instead for using a massive framework like react.

[Skip to main content](#)[Log In](#)**joyrexj9** • 20 days ago • Edited 20 days ago

Long time Go dev and wanted to give HTMX a spin so I went down the Go + Echo + html/template route.

After spending a lot of time with SPA and JS frameworks (React and Vue) I have to say I love working with Go and HTMX. It takes a while to unlearn things and go back to thinking of everything being on the server, but the simplicity and elegance really shines through.

Haven't tried using templ, but it looks a little framework-y for my tastes. The thing I wanted to use Go + HTMX to get away from

⬆ 20 ⬇ Reply ⬆ Share ...

**Aspect3120** • 20 days ago

Totally agree! My first time I tried templ, I was brand new to go and I couldn't get it, now I've been doing go for a while and I think I could, but I simply don't want to lol. I am trying to avoid react and those silly frameworks and html/templates work so perfect

⬆ 8 ⬇ Reply ⬆ Share ...

⊕ 2 more replies

**BorinAxebearer** • 20 days ago

I created a small project using this stack

Repo: <https://github.com/msyavuz/Lua-Capture>

⬆ 9 ⬇ Reply ⬆ Share ...

**BenPate5280** • 20 days ago

I'm doing this, too (with a few extra steps) and it's fantastic. Go's templates work beautifully with htmx. Throw in some extra logic for GET vs POST requests, and your application is nearly done already.

OP, are you building something public that you can show off?

⬆ 14 ⬇ Reply ⬆ Share ...

⊕ 4 more replies

**kynrai** • 20 days ago

Going all in on this as my stack of the future. Pushing it internally and to public sector UK gov organisations.

Main benefit is the entire stack is backwards compatible, few dependencies, super easy to upgrade with new go versions, super easy to containerise and a great Dev experience

⬆ 5 ⬇ Reply ⬆ Share ...

**Mental-Sun5667** • 20 days ago

[Skip to main content](#)[Log In](#)[+ 6 more replies](#)**InterestingPatient49** • 20 days ago

Seriously, all this htmx trend... Does anyone have seen a real project using it? Like a big, complex one and no It just TODO apps?



7



Reply



Share

**teodorkostadinov** • 20 days ago

I am also interested to find out.

This is the only video I have found for HTMX used in prod <https://m.youtube.com/watch?v=3GObi93tjZI>



2



Reply



Share

**Savagor** • 20 days ago

It's very early stages, but this guy is building a web framework with all batteries included and it uses htmx and templ. Maybe you want to keep an eye on it! <https://github.com/anthdm/slick>



5



Reply



Share

[+ 1 more reply](#)**International\_Cap909** • 20 days ago

I'm using stlib go templates + alpinejs.

But I'm curious to test this stack. Seems to be great.



3



Reply



Share

**bluebugs** • 20 days ago

Yes, it is fast to get going and have a modern web application working as a backend developer. Specially useful when all the front-end engineer are busy building the product and someone need to provide tools for everyone.

Another tool I am adding to this list is playwright-go which allow to run end to end tests from within go test. This allow me to check the value I expect are in the dom and that at least a few screen display the pixels I expect. Make it easy to rely on dependabot after that to keep things up to date and not breaking.



3



Reply



Share

**Tha-Kee** • 20 days ago

Look at this , awesome : <https://www.youtube.com/watch?v=2KyZJVQFa5M>



2



Reply



Share

[+ 1 more reply](#)

[Skip to main content](#)[Log In](#)

Coming from devops/backend/data engineering world, this combo is a Godsend.

I started building our site using Next.js and React and I hated it. I like knowing what every file means / does (as much as possible); I already loved using Go, so the Next.js/React world was not a great experience (for me — I'm sure others are very comfortable with it).

Also I like the ability to deploy anywhere, and Next.js is *almost* pointless outside of Vercel imo.

I would highly recommend this approach and I hope to see more folks using it in the future. It's fast, can be done by any Go engineer (ostensibly), and it's really a back-to-basics as far as the web goes. JS frameworks / Django / Rails abstract too much away for my own liking.

\*This is not investment advice. Do what's best for you and your org; make prudent decisions.

EDIT: I also work with Flutter/Dart for the mobile side of our platform rather than Swift/Kotlin/React Native, so Go + HTMX is a particularly good fit in my case.

⊖ ↑ 2 ↓ [Reply](#) [Share](#) ...



**greasyjamici** • 10 days ago

It sounds like you primarily do web dev with Go these days, but have you done much with Go for data engineering? If so, how do you find it for data engineering in general? Do you think it's as productive for ETL jobs, simple REST APIs, etc. like other languages typically used in DE (namely Python)?

↑ 2 ↓ [Reply](#) [Share](#) ...

⊕ 3 more replies



**kwhahn** • 19 days ago

It is so good and makes development so much easier, faster and less complex. The problem is the rehab from all the years with JS frontend frameworks. Templ is absolutely amazing and also makes deployment much easier.

Will definitely try to utilize the stack more.

⊖ ↑ 1 ↓ [Reply](#) [Share](#) ...



**NiceCelery499** OP • 18 days ago

The primeagen says that 2024 is the year of htmx so leeeet goooo hahahah

I hope he's right

↑ 1 ↓ [Reply](#) [Share](#) ...

[View more comments](#)

## More posts you may like

Skip to main content



Log In

Part 4. ASP.NET 5.0 JWT Authentication



69 upvotes · 11 comments



Linus on Rust in the Linux kernel (December 2023)



216 upvotes · 83 comments



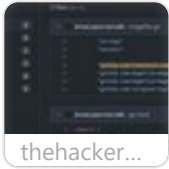
John Carmack telling NASA Engineers that Rocket Science is simple compared to Graphics Programming



275 upvotes · 50 comments



15,000 Go Module Repositories on GitHub Vulnerable to Repojacking Attack



109 upvotes · 41 comments

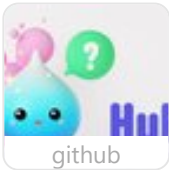


I'm coming from Java and I have been told that I'm writing go like I'm writing Java. Basically creating structs, injecting fields, and attaching methods. What else can I do?

93 upvotes · 40 comments



Huh: a simple, powerful library for building terminal forms (written in Go)



200 upvotes · 36 comments



My Concurrent Programming book is finally PUBLISHED!!!

350 upvotes · 48 comments

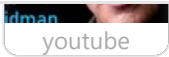


Go 1.21.5 is released

112 upvotes · 13 comments



Log In



139 upvotes · 120 comments

 r/golang

Go, nil, panic, and the billion dollar mistake

916 upvotes · 397 comments

 r/golang

When should I use pointers?

93 upvotes · 62 comments