

Creating Web Apps Using Go

My Intro to Frontend Development
with Air, Templ, and HTMX, oh my!

Paul Balogh, @javaducky

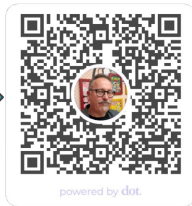


Disclaimer

Paul is **not** here to sway you to one **product** over another, nor am I here to persuade you into a specific **paradigm** over another.

He is interested in experimentation and learning.

Forward complaints to





Act I

So, we have this application...

The Challenge

- An application written primarily in Perl CGI scripts
- Used to monitor status for 1,000s of services
- Multiple components covering web and probes
- Want to migrate to AWS with Aurora Global database
- Afforded an opportunity for a Proof-of-Concept for conversion to Go
- Been running since...



The Goal

- A **single binary** written in Go to replace:
 - Perl web application (read-only) for status dashboard
 - Executable Jar-file and multiple Bash+Python scripts utilized for synchronizing configuration
 - Perl script used to probe remote services



The Why

- This may be the **only** thing we have written in Perl

No one here is creating anything new with it

- **Explore** the **viability** of creating new services with Go

We have more technical depth with Go than Perl

- This application is not customer-facing, so **low risk** should things not pan out



Act II

A Brave New World



What the stack?

Based on *whispers* on the internet

- **Air** - <https://github.com/cosmtrek/air> for live-reload
- **Templ** - <https://templ.guide> for HTML templating
- **HTMX** - <https://htmx.org/> for dynamic page updates
- **chi** - <https://go-chi.io/> for HTTP routing
- NO ORMs...keep it simple

Let's Review the Code

Let's take a look at setting up your environment for building the project. Steps include:

- Pull down the source code
- Install Air and Templ

Let's Review the Code

YOU CAN NOT SHARE OUR CODE!!!

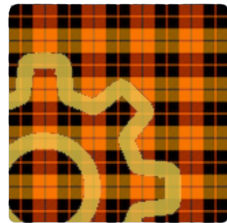
Let's take a look at setting up your environment for building the project. Steps include:

- Pull down the source code
- Install Air and Templ



Let's Review the Code

- An MIT-licensed, open source example **microservice** with RESTful APIs
- Backed by a relational **database**
- **Many** implementations; **same** API requirements
- Inspired by **TodoMVC** (<https://todomvc.com/>)
- **Compatibility** (Contract) testing with k6
- Started *way back* in **2019**!



Setting up Air

- Straight-forward setup
 - Install using ``go get``
 - Run ``air init`` to generate basic config
 - Modify from there
- I added a ``make develop`` target to start

<https://github.com/weesvc/weesvc-gorilla/commit/b2546fda6e>

THIS SLIDE INTENTIONALLY LEFT BLANK

Setting up Templ

- Straight-forward setup once again
 - Install using ``go get``
 - Run ``air init`` to generate basic config
 - Modify from there
- I updated my ``make develop`` target to start

<https://github.com/weesvc/weesvc-gorilla/commit/0570d9eee1>

THIS SLIDE INTENTIONALLY LEFT BLANK

</>htmx

What's the fuss?

- A single 48k (14k gzip'd) minified JavaScript file
- Enables backend developers to make cool UI

ThePrimeagen says it's cool!!! And he's right...

<https://github.com/weesvc/weesvc-gorilla/commit/a3e3552b05>

“ ”

But, I didn't use it for work...yet ;)



THIS SLIDE INTENTIONALLY LEFT BLANK



Act III

The verdict...



“ ”

I really do like it.



Taking note

- Turned off `'goimports'` as Templ formatted a little differently
- Occasional confusion between Air and Templ watchers...needed an explicit `'templ generate .'`
- Code assist with GoLand plugin didn't always work; unsure of VSC
- Air is great for fast iteration of UI and code, but **not** great for debugging

Taking note

- Organize Templ and HTTP Handlers with HTMX in mind; smaller “partials” of HTML code for swaps
 - Do not underestimate the power of Cobra/Viper integration for command-line options/config
- and finally...

Friends don't let friends write Perl!

Evaluation



- While new-ish to frontend development, **productivity was very fast**

Thanks to Go's fast compiling and the Air and Templ file watchers!

- Creating web applications with Go is **totally viable**, though I wouldn't do this for large UI projects

Frontend pros can still use React with a Go/Templ/HTMX backend!

cloud gnome



Thank you!

Connect with Paul as
[@javaducky](#) or [linkedin/in/pabalogh](#)



KEEPING WEEDS OUT OF YOUR CLOUD NATIVE GARDEN

THIS SLIDE INTENTIONALLY LEFT BLANK