

PHP, TWIG, AND KEEPING IT FRESH...

TEMPLATING REST APIS

[N8FINCH.COM/TALKS](https://n8finch.com/talks)

@N8FINCH

NATE FINCH... WHO??

- ▶ WP 2012, 2014 went full-time
- ▶ Previously: [toptal](#), [Codeable](#), [10up](#)
- ▶ VP of Development at [The Digital Ring](#)
- ▶ Texas >> Kansas >> Spain (a little time in Morocco) >> France >> Illinois >> S. Korea >> and now Madison
- ▶ *"I'd be a professional student if I could..."*
- ▶ Married w/ 2 yo daughter & #2 coming soon!



WHAT IS TEMPLATING?

SOME PRELIMINARY THOUGHTS

- ▶ A template is simply a text file.
- ▶ A template contains variables or expressions, which get replaced with values when the template is evaluated, and tags, which control the logic of the template.
- ▶ Creating partials and layouts with slots into which data can be substituted.
- ▶ Templates across a site typically have the same structure, but the content is variable.
- ▶ Examples from ever day life.

PHP STARTED AS A TEMPLATING LANGUAGE

"I wanted a simple templating language that was built into the web server... so the web server could do intelligent substitutions... I saw PHP as a very thin templating system on top of business logic written in C... And I provided a very nice API... for writing all your business logic in C, and then you would just do the front end stuff-- the actual display logic--in the templating language... The world disagreed with me, nobody wanted to write in C... People could add one little thing to their HTML page, reload their browser, and hey! It worked! They didn't have to compile anything...It was just easy..."

Rasmus Lerdorf,
@ WeAreDevelopers Conference 2017

WHY USE TEMPLATING?

PLENTY OF REASONS

- ▶ Dynamic content vs. static files.
- ▶ Partials, reusable pieces of a site (e.g. header, footer, content, sidebar, tables, forms, etc...)
- ▶ Avoid repetition, code is in one place
- ▶ Built in security controls, like escaping (late!) ("When working with data, validate and sanitize early, escape late.")
- ▶ Keep "business logic" out of the views... kinda... basically...probably...

**WHERE DO I GET THIS
“TEMPLATING” YOU
SPEAK OF?**

PLENTY OF OPTIONS

- ▶ [PHP](#) (yep! just use PHP like it was originally intended), this is what WordPress does.
- ▶ [Twig](#)
- ▶ [Blade](#)
- ▶ [Smarty](#)
- ▶ [Mustache](#) || [Handlebars](#)
- ▶ [Dwoo](#) (🙄)



A COMPARISON

```

<h1><?php echo $twig_array['name']; ?>'s Todos </h1>

<ul>
  <?php foreach ($twig_array['todos'] as $todo) : ?>
    <li>
      <label for="todo-<?php echo $todo['id']; ?>">
        <input id="todo-<?php echo $todo['id']; ?>" name="todo-<?php echo $todo['id']; ?>" type="checkbox" value="<?php echo $todo['id']; ?>" <?php echo $todo['completed'] ? 'checked' : '';?> /><?php echo $todo['title']; ?>
      </label>
    </li>
  <?php endforeach; ?>
</ul>

```

PHP

```

<h1>{{ name | upper }}'s 2-Dos </h1>

<ul>
  {% for item in todos %}
    <li>
      <label for="todo-{{ item.id }}">
        <input id="todo-{{ item.id }}" name="todo-{{ item.id }}" type="checkbox" value="{{ item.id }}" {{ item.completed ? 'checked' : '' }} />{{ item.title }}
      </label>
    </li>
  {% endfor %}
</ul>

```

TWIG

**SOME EXAMPLES OF HOW
TEMPLATING IS USED IN A
FEW FRAMEWORKS**

EXAMPLES

- ▶ WordPress (template tags and conditionals, escaping data, could talk about validating and sanitizing as well)
- ▶ Symfony (Twig)
- ▶ Laravel (Blade)

**LET'S CHECK
THIS OUT!**

HANDS ON: THE REPOS

- ▶ PHP and Twig comparison:
 - ▶ <https://github.com/n8finch/twig-template-testing>
- ▶ Twig and Routing setup for REST API:
 - ▶ <https://github.com/n8finch/php-templating-matc>

HANDS ON: PHP VS. TWIG

- ▶ Basic example of a todo list, checking out
- ▶ What are the differences between PHP and Twig?
- ▶ What extra "sugar" does Twig provide out of the box?
- ▶ Do you have a preference for either based on:
 - ▶ functionality
 - ▶ sugar
 - ▶ look and feel

HANDS ON: TEMPLATING THE REST API

- ▶ Extend the REST API to use an interface:
 - ▶ Routing with switch (or recreating the wheel)
 - ▶ Work out templates (header, footer, main body, etc.)
 - ▶ Form setup based on routing
 - ▶ Extend with @method
 - ▶ Extend with @csrf token

QUESTIONS???

NO?? OK, COOL, THANKS!

IF YOU THOUGHT THIS WAS FUN...



WE'RE ALWAYS LOOKING FOR FUN FOLKS!