



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

## CTRL+CLICK CAST #11 CMS Markup & Templating with Allison Wagner

[Music]

**Lea Alcantara:** You are listening to CTRL+CLICK CAST. We inspect the web for you! Today we're talking about markup and templates for content management systems with special guest, Allison Wagner. I'm your host, Lea Alcantara, and I'm joined by my fab co-host:

**Emily Lewis:** Emily Lewis.

**Lea Alcantara:** This episode is brought to you by Squarespace, the all-in-one platform that makes it fast and easy to create your own professional website or online portfolio. For a free trial and 10% off your first purchase, go to [squarespace.com/click](https://squarespace.com/click) and use the offer code CLICK.

**Emily Lewis:** CTRL+CLICK would also like to thank [Pixel & Tonic](#) for being our major sponsor of the year. [Music ends] Hey Lea, how are you doing?

**Lea Alcantara:** Pretty good. How are you?

**Emily Lewis:** Not too bad, so I know you've only been in Seattle about six months, but did you catch the football game yesterday?

**Lea Alcantara:** I caught part of it. I was actually doing a wine and dine thing with a bunch of girlfriends.

**Emily Lewis:** [Laughs] It sounds awesome.

**Lea Alcantara:** So we missed a majority of it, but we couldn't really miss it because everywhere we went, people were in Seahawk's gear and then how we kept up with the scores essentially when there was a cheer. [Laughs]



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** [Laughs]

**Lea Alcantara:** And then when we went out for dinner, there was like one TV in that restaurant and we didn't know what was going on until suddenly there was like this huge roar.

**Emily Lewis:** [Laughs]

**Lea Alcantara:** A roar, and we're really like, "I guess we won." [Laughs]

**Emily Lewis:** That's cool. I've actually never been in the city of a winning team as like an adult. I mean, I grew up in the Washington, DC area and the Redskins went into the SuperBowl a few times back then, but that must be really exciting with that sort of energy.

**Lea Alcantara:** Well, I have mixed feelings because, well, I'm so Canadian. The only sport I really know is hockey. [Laughs]

**Emily Lewis:** [Laughs]

**Lea Alcantara:** So it's kind of like football, yay, home team, you know? [Laughs] But like I'm happy for Seattle and I'm happy for people that are fans of the game and all that fun stuff but...

**Emily Lewis:** Well, and I'm sure like some of your new neighbors and friends in the area, you're going to get invited to at least one really good SuperBowl party.

**Lea Alcantara:** Yeah, and I'm actually looking forward to the food more than even the game. It's just like, "What will we prepare for SuperBowl party?" [Laughs]

**Emily Lewis:** Well, and do you know about the whole commercials at SuperBowl time that's in there?

**Lea Alcantara:** Oh yeah.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** Okay.

**Lea Alcantara:** Yeah, of course, because they're so massive and then they become viral sometimes too.

**Emily Lewis:** Right.

**Lea Alcantara:** So yeah, for sure.

**Emily Lewis:** Well, that should be fun then.

**Lea Alcantara:** Yeah, they'll be interesting, but it will be my first like real, huge sporting kind of experience in the States, so we'll see how it goes. [Laughs]

**Emily Lewis:** [Laughs] Well, let's go ahead and get into some news in the world of content management systems. To start with, Perch released an update, Version 2.4, that fixes some bugs, but also introduces a few new features. One of these features is interesting. I really want to get my hands on Perch so I can see how some of these works, but it's something called Template Dividers, and with your template tags, you can generate some markup to split a template into sections without actually, from what I can understand, writing the markup yourself. It sort of just generates it directly for you in the user interface, so that's kind of interesting, especially concerning what we're going to be talking about today.

**Lea Alcantara:** Hmm, very cool, and in other CMS news, Craft has released another update to its contact form plugin to support multiple email addresses.

**Emily Lewis:** And I'm not sure if this is new or just new to me, but I saw a link on the #eecms for this link on EllisLab site. It's [ellislab.com/expressionengine/why](http://ellislab.com/expressionengine/why), and it's like a little marketing slideshow resource thingy that highlights the reasons to use ExpressionEngine.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Lea Alcantara:** [Agrees]

**Emily Lewis:** This was the first time I came across it, and I thought it was nice because I want to see EllisLab doing more things like this.

**Lea Alcantara:** Yeah, absolutely.

**Emily Lewis:** I feel like they just need to do more marketing to build name recognition for ExpressionEngine just based on the number of RFPs and job requests that we see, almost none ever mention ExpressionEngine, and so there's already a hurdle that we have to get over to educate our clients about what other options are out there other than WordPress or something like that.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** But I will say I was also a little confused about this tool because it seems a little multiple personality in the sense that some of the points are relevant to EE devs like you and me, and some are kind of relevant to our clients who are making the decision on what tool they might want to use or what kind of agency they want to go with and what that agency might work with.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** And I thought that was sort of weird because I wouldn't want to send that link to a prospective client because I don't think half of the points are relevant to their perspective. So I'm kind of torn on it, but it did remind me of that tool that Michael Boyink and Marcus Neto put together.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** Was it early last year or late 2012? But it was basically ExpressionEngine at a glance, we'll have the link in our show notes, and they say it's aimed at newbie developers, but I really think



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

it's a much more effective resource for a prospective client to get a sense of all of the options that they could had if ExpressionEngine is the tool for their CMS.

**Lea Alcantara:** Yeah, I think it's always a balancing act over, well, who's the audience here.

**Emily Lewis:** [Agrees]

**Lea Alcantara:** Because the way you speak to a client, but which client are you talking about? Like is the client a more technically inclined client, or are you talking towards marketing type of person, or a businessperson that just simply wants to understand how flexible the system is.

**Emily Lewis:** [Agrees]

**Lea Alcantara:** And that's all the news for today. If we overlook news about a CMS you favor or are interested in, let us know and we'll get it on our radar for feature episodes.

**Emily Lewis:** All right, so I'm really excited about today's episode because we're going to talk about markup, which is my favorite topic. Of absolutely anything related to web development, I love HTML the most.

**Lea Alcantara:** [Laughs]

**Emily Lewis:** We have a special guest, Allison Wagner joining us to talk about the markup and templates for content management systems. Allison is a front-end developer for Happy Cog where she specializes in the art of abstracting design systems from comps to code. When she's not coding, she's sharing her knowledge about coding, speaking at conferences, writing for Web Standards Sherpa and Happy Cog's Cognition, and teaching for GirlDevelopIt!. Welcome to the show, Allison. Thanks for joining us.

**Allison Wagner:** Hi, how are you both doing?



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** Great!

**Lea Alcantara:** Pretty good. So Allison, can you tell our listeners a bit more about yourself?

**Allison Wagner:** Sure. I work in the Philly office of Happy Cog. I have a house here really into music. I'm actually going to a concert on Wednesday to go and see Gene Clark's *No Other*, a super group of Beach House and Grizzly Bear and like members of a few other awesome alternative kind of bands. It's like a flea market so that's a little bit more about me. I'm so happy to be here. Yay!  
[Laughs]

**Lea Alcantara:** [Laughs]

**Emily Lewis:** [Laughs]

**Lea Alcantara:** Hurray!

**Emily Lewis:** Well, we're really excited you're able to join us because you had to have your wisdom teeth extracted last week.

**Allison Wagner:** [Laughs] Oh, yes, I did. Yeah, that was the worst. [Laughs]

**Lea Alcantara:** [Laughs]

**Emily Lewis:** [Laughs]

**Allison Wagner:** But I'm on the up and up now, so I'm doing much better, thank you. [Laughs]

**Emily Lewis:** So before we get into some of the specific questions that we have about markup for content management systems, I kind of want to get a peek behind the curtain of Happy Cog in the sense of how do you work, like you work as a developer for them. Do you work in teams of two or more people, or are you independent where you're just handed projects and you just build the front end and that just gets hands off to someone else, or how does it all work?

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** Yeah, so I've been with Happy Cog now for four years or four and a half years, and the company has gone through quite a genesis, I think, over the last four years, specifically the last year. When I started, I think there were like 11 or 12 people, and now we're up to 30 and we used to kind of sell the company that when you work with Happy Cog, you work with the entire team. Since then we've kind of split into small teams of five or six people each and they're actually cross-office. We have several offices, one in Austin, one in New York and then one in Philly, and so there are about five to six team members that tackle projects individually.

It's like they're a little independent company, which is nice, because there's a lot of opportunity for innovation and we can try new things within each kind of small group and then we have lunch and learns, that kind of deal, and then share our findings. So there is a lot of opportunity there to try new things and have a cross-office collaboration and all that kind of deal. So we are involved from the very beginning, so after the sales team kind of secures the job, then we have a sales-to-project transition meeting and we go right into the stakeholder interviews, and we're super collaborative. I'm involved in stakeholder interviews and I'm doing requirements gathering and that kind of a thing and helping with wireframing and all that kind of jazz.

**Emily Lewis:** Oh, okay.

**Allison Wagner:** So yeah, I'm not just in a developer hole kind of a thing, like all by myself or like sad. I'm working with the team and being involved in all phases so it's really an interesting kind of newer way to approach templates and development in general. So yeah, I'm really happy with the direction that we're moving.

**Emily Lewis:** Cool. So do you find yourself working with different people for different projects? Or you're kind of always working with the same people each time within your own team or office, I guess?

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Timestamp:** 00:09:46

**Allison Wagner:** Yeah, right. Well, so I'm generally working with the same people. This is all very new. We really just kind of started it in September. There was a talk that we would do team structure for a year and then reevaluate and kind of see what the pros and cons were, and then possibly shuffle people around base on strengths and what they're actually interested in working on. Because teams are kind of curtailed right now for what it is they actually want to focus on, and people are strategically put into teams based on their strengths and weaknesses.

**Emily Lewis:** [Agrees]

**Allison Wagner:** And hoping that people will come together and kind of fill in the blanks where necessary. But then also, there are always going to be times where there's overlap or someone is out unexpectedly, that kind of a deal, so I still pop into other projects and help out where I can, with QA, that kind of a deal. So I'm still working with other people at the company when I'm entirely isolated, but we do primarily work with our small group.

**Emily Lewis:** Cool.

**Allison Wagner:** Yeah.

**Emily Lewis:** Now, how did you get into web design and development? Was that what you went to school for?

**Allison Wagner:** Not really, no. I thought I was going to be a pharmaceutical sales rep. [Laughs]

**Emily Lewis:** [Laughs]

**Lea Alcantara:** Wow!

**Allison Wagner:** Yeah, my whole family...

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Lea Alcantara:** That's what my mom was.

**Allison Wagner:** Yeah, my whole family is in pharmaceutical, like my mom and dad, my uncle, my sister, everybody, and so I thought I was going to go that route. I went to school for Biology and then I didn't like it, and I've always kind of thought I'd do journalism or something more creative. I always kind of have a creative bent to me, so I switched over to school communications and I went to Temple University, and they had just started an advertising program. So I think I was in the second class to graduate from the advertising program.

**Emily Lewis:** Oh.

**Allison Wagner:** And in my senior year, there was a class. It was like interactive design or something like that, and the professor is Jade Snyder who worked at Saatchi & Saatchi Interactive and she kind of introduced us to the idea that people make websites and the process behind it, and I just really kind of fell in love with the idea and I downloaded Photoshop and took some Lynda classes on Photoshop, and being able to manipulate graphics and imagery was super awesome.

So then I started designing a site for myself. I was making jewelry at the time so I wanted to make a site for my jewelry making, and after I designed it, I just kind of started to build it as well. I mean, I was in Dreamweaver doing the drag-and-drop kind of like super WYSIWYG development at that point, but it was a nice way to get my feet wet and I just kind of fell in love with the process of translating design into development, abstracting it into the values and all that kind of thing, and that was really interesting to me. So once I built my own site, I got hired right out of school and I was at a mid-sized ad agency over in Jersey doing email blasts and Flash animations, you know, pay my dues.

**Emily Lewis:** [Laughs]



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** But while I was there I was also doing tutorials and that kind of stuff and I was reading a lot of *CSS Tricks* and *Smashing Mag* and continuing with the Lynda and there are all those PSD to HTML tutorials, that kind of stuff, so I was just spending a lot of time trying to teach myself best practices and get into just specifically web development. I thought I was actually going to be a web designer at first and then I found myself much more suited for dev.

But yes, so that was kind of it. I was there for a year and then I went to Happy Cog, and I thought I was actually going to be an intern at Happy Cog, but they liked my portfolio I had taken on a project where my dad was on the board for his community, but I don't know, whatever, and they wanted a website so I learned WordPress so that I could build them the site and I think they were impressed by that.

When people say "self taught" in web development, it depends on how isolated you are. I was not terribly isolated. I was working with the team at Happy Cog so I was still continuing to learn and I'm still learning to this day, so it's a balance of having this snazzy or....

**Emily Lewis:** The initiative...

**Allison Wagner:** Yeah, thank you. Yeah, to like read up and always like stay thirsty for knowledge or whatever and then also working with the team and sharing your findings, that kind of a thing. So yeah, now here I am four and a half years later and I absolutely love it.

**Emily Lewis:** That's awesome.

**Lea Alcantara:** So speaking of learning new things, how did that transition to picking up content management systems and learning more. You mentioned WordPress, was that the first system you started fiddling with?



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** Yeah, yeah. So I started there, and there was just a ton of documentation on WordPress, so I think that's a really good starter CMS for people just because that there are so much hand holding, or not hand holding, but there are so much or so many...

**Lea Alcantara:** Resources.

**Allison Wagner:** Yeah, with like awesome tutorials and stuff online that you can just type in to Google exactly what you want to learn and somebody has written an article on it. So I started there, and eventually I taught myself ExpressionEngine because that would be the primary CMS that we were using at Happy Cog because I think previously, CMSs in general were like a black box to me, so I would write HTML because I was just primarily doing static websites, and with all this unknown around a CMS, it was like super scary to me, and I didn't really know what I was scared of. I guess that's probably part of the fear.

But once I started to learn about variables and how that plugs content into your static HTML and how CMSs actually work the databases and that kind of thing and demystifying that, it took a lot of the hesitation I had around the CMS topic in general and kind of it made me more comfortable with it, and that confidence carries through into your work on how you approach development in general.

**Emily Lewis:** Yeah, and it also just makes, especially if you're working with other people, the conversation is easier. You can use the right terms to convey something when you have the context of how the CMS interacts with your markup. I remember that point from myself when I learned ExpressionEngine for the first time, there was that light bulb moment.

**Allison Wagner:** Yeah, I read it. I think Ryan Irelan who actually works for Happy Cog wrote a book on learning ExpressionEngine and reading that was huge for me because it was just clear, and I was just kind of reading through it and I remember I had like so many aha moments. Just in the first



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

chapter alone, it's like, "Oh okay, this make sense. You just put this variable here and then it prints out whatever they've added into the admin." It just started to make sense about how static templates and a CMS interact.

**Emily Lewis:** [Agrees]

**Allison Wagner:** And it helped me tremendously. I think just kind of diving in and learning things that you're even just on a surface level, if you're not going to go whole hog into a project, but at least familiarizing yourself with the topic helps tremendously. At least for me, because if it's a confidence thing and you're hesitant to talk about it with your coworkers because you don't want to reveal how little you know, that kind of thing?

**Emily Lewis:** Yeah.

**Allison Wagner:** I don't want to have to negotiate that kind of personal politic. I just want to be able to feel confident about something, so that was really awesome. Learning ExpressionEngine, I'd say even more so than WordPress, it was huge for me being able to build a full site from start to finish and untouched kind of thing from outside was pretty big, and I did that a lot in my freelance career where I use ExpressionEngine a lot.

**Emily Lewis:** Oh, so you don't just do like the templates, but you also would do the development in the CMS...

**Allison Wagner:** Yeah.

**Emily Lewis:** Building out your custom fields and channels and things like that?

**Allison Wagner:** Exactly, yeah, yeah, yeah.



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** So you've worked with WordPress, ExpressionEngine, anything else? Even just testing something or experimenting?

**Allison Wagner:** Well, I've written templates for Drupal, but I haven't done any Drupal dev, and then also Craft as well, but I haven't done any Craft yet, but I've just done templates for them, and I'm sure I've done templates for other systems as well. But as far as being involved in the front end to back end, that would be WordPress and ExpressionEngine.

**Emily Lewis:** So amongst those three, were there any unique approaches you needed to follow with your markup to support, let's say, the Drupal templates, or could you just write straight up markup and you trusted that your CMS dev just took it from there?

**Allison Wagner:** Well, ExpressionEngine is really nice because I feel like it's minimally invasive in templates and pretty much what write is what's going to appear on the front end for the finished product. Drupal, I've worked on two Drupal projects, and the first one, we learned a lot of lessons just as far as our process goes. So I would work with the back-end developer and we were typically working Waterfall at the time so I would complete templates entirely, and then I handed them off to the back-end developer for Drupal, and we found that it probably wasn't the best approach because Drupal does kind of have a lot of front-end implications. They do add classes and markup and that kind of thing. It was much more of a beast to work with.

I understand that there's a lot of positives with working with Drupal, but as far as the front-end implications goes specifically, there's a lot more you have to kind of take into account. It doesn't just leave your templates as is or as clean, so just dealing with their own CSS that they added as well, so we made changes to the process moving forward. If we're going to use Drupal, just make sure that



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

there's time for me to go back in after the back-end dev has completed coding or is midway through coding or content is being entered to make sure that things are kind of looking correct and that the Drupal CSS is an overwriting things in some instances because that happened, and there are some complications involved. But then the next Drupal project that we worked on went a lot smoother because lessons were learned.

**Emily Lewis:** [Agrees]

**Allison Wagner:** And we just accounted for it in terms of time.

**Lea Alcantara:** So when you're talking about accounting for it though, can you talk about a little bit more specific because Waterfall is pretty straightforward, like you finish your templates then hand it off. But now it sounds like it's a little bit more collaborative. How does that work when what you're still building is still ongoing, the CSS is still in the process and then he's still trying to plug in the Drupal code or whatever code? How do you communicate where to stop and send the other dev something?

**Timestamp:** 00:19:56

**Allison Wagner:** Yeah, so I would like complete a suite of templates, maybe three or four templates, and then pass them to him, and this whole time, communication is huge and planning is huge, so just making sure that I was being resourceful specifically because I work in an organization. I'm not just an army of one or whatever. I'm making sure that my time, that I have time every week to go back and check and make sure that things are being implemented correctly and classes are being translated over and that things aren't getting overwritten and that the classes that are applied are correct, that kind of a deal. I'm just kind of making sure that we're accounting for all the variables and things that happen when you're implementing into Drupal.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

So I think that with ExpressionEngine when I was done with templates, and done, meaning that I had written all the HTML, CSS, JavaScript and then QA-ed across all the browsers and devices, that kind of thing, you could traditionally say, “Okay, then I’m done with this project. As a front-end developer, I can move on to the next project.” But with Drupal specifically, you’re not really done. You have to kind of stay in the loop with things and just kind of check back, and we use Sifter which is a bug-tracking system.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** [Agrees]

**Allison Wagner:** So tickets would be logged there, or as pages were completed or templates were completed, there would be a ticket logged, then I would have to go back and check to make sure that things kind of translated nicely, and it’s just making sure that communication is key, and we all work in the office here so Anthony [Colangelo] who I work with on Black Hills which was our Drupal project, he would just tap me on the shoulder kind of a deal and I would roll my chair over and we’d look at something together as to why it’s not printing correctly and typically solve the problem very fast, but just making sure that you have a good relationship with your coworkers, which is huge, obviously.

**Emily Lewis:** So when it comes to a CMS, maybe not so much right now while you’re working with Happy Cog, although maybe you are kind of have a window into the sales process, but even when you were freelancing, how the templating works for a CMS? Did that influence what CMS was chosen for a project?



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** Yeah, I think that, well, any CMS that spits out junk markup or inaccessible markup or less than ideal markup that is unchanging and hard to control would take a lot of effort on our part to overwrite the default system, it would probably be thrown out the window immediately.

**Lea Alcantara:** [Agrees]

**Allison Wagner:** Because we pride ourselves on accessible content, and so we have three or four that we will use because we actually do something called a tech approach where we'll go through and kind of test each or rank each CMS against a set of criteria and make sure exactly all of the requirements are being met.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** [Agrees]

**Allison Wagner:** This is after the stakeholder interviews and that kind of thing, so we kind of know the client a little bit and what their expectation is and what they're looking for, and we are typically doing very content-heavy sites, not so much app sites.

**Lea Alcantara:** So after you've chosen a CMS to work with or the company you're working with has chosen a CMS to work with, the next step obviously is to start working with the project. One thing that I've heard of when I talk to other devs about templating is that DRY or DRY process with front-end templates. What's your thoughts about that process when you're doing any front-end coding for a CMS?

**Allison Wagner:** Yeah, I think that the best approach is your own personal or an approach that you're very comfortable with.

**Lea Alcantara:** [Agrees]

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** So just have super sound starter files with conventions that you're familiar with as far as classing and that kind of thing goes, and then just kind of really focus on making sure that those are super solid and sound so that you can use that over and over again, and that kind of helps you create your system and you know going into templating what your classes are going to look like and how things are kind of the art is going to be set up. So I try to keep things in a healthy balance. You don't want to be too dry and then maybe have like a body class and have that override everything and then you're getting super specific and then you're fighting with yourself that you're not using CSS to its fullest. You're not using the cascade or inheritance.

I like to keep things super modular so I will class a `<div>` or an `<aside>` or a `<section>`. If it's a callout or something, I will do a class of callout and then the `<h2>`, `<h3>`, `<h4>` that appear inside it will take a default style and the `<p>` will take a default style and that kind of thing so I'm kind of classing based on the module instead of the page itself, and that kind of helps keep things low level.

**Lea Alcantara:** [Agrees]

**Allison Wagner:** As a general rule, I try not to go more than three selectors in a string because I feel like once you're there, four just feels like too much to me.

**Lea Alcantara:** Sure.

**Allison Wagner:** And you're not really using the system, or you're not using CSS for how it could be. It just starts to take up too much time and that kind of deal. So keeping classes like one to two on a module, if there's a "callout" and then "callout-alt" or something because `.callout-alt` has a border and a shadow or like the heading color is different, targeting that way based on the module instead of on individual elements inside of it. But then don't make it hard for yourself. If you have to add a class, add a class, you know?



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Lea Alcantara:** [Agrees]

**Allison Wagner:** You don't spend too much time moaning over whether or not you should add a class here or not. But if you need to, then you need to, and you can always go back and change it if necessary. But this also speaks to just kind of planning in general when you have your design system kind of all set out in front of you, making sure that you're scanning all of the templates and looking for the smallest units of design and then coding them and then using the default styles that you've already written to your advantage. Just always looking for things that repeat, text-transform: uppercase on all metadata and just make sure that you're looking for those patterns and writing them in early and then overwriting them when necessary.

**Lea Alcantara:** [Agrees]

**Allison Wagner:** But that sound base is going to support a sound design system.

**Emily Lewis:** So those base resources or I think you referred to them as your starter files, from where you are at Happy Cog, is it something that's like the whole team uses the same kind or each developer has their own set of what naming conventions and things make sense to them?

**Allison Wagner:** Yeah, we have Happy Cog starter files, which we traditionally use and then update, and we actually just updated to Grunt away from CodeKit, so we're always changing them and updating them based on what's going on, what's the best, I guess, approach, or how we feel things are moving in a certain direction, and it's nice because I think at this point, gosh, we probably have ten developers, maybe twelve developers, so when we're all using the same starter files and then making updates as necessary, and then committing them up to GitHub with the comment as to why you changed the started file and then everyone kind of learns from that.

**Emily Lewis:** [Agrees]

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** So it's a shared resource that we use across the projects. There has been times when we're doing internal projects, that kind of thing, where people will kind of go their own way as an opportunity to experiment, which I think is smart, and then port those same ideas back to our starter files and kind of try to help the whole team out as a whole.

**Emily Lewis:** Right.

**Allison Wagner:** We'll have dev meetings and talk about what's going on in the development community and what are some things that are worth trying and what you've learned and what you don't like, that kind of a deal, and that's very helpful as well, but the starter files are crucial, and I have my own personal starter files that I use as well, but that's a freelancing side. They are far more scaled back.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** [Agrees]

**Allison Wagner:** But the Happy Cog starter files are pretty robust and awesome as well. I'm a big fan of them. We're always revisiting them and discussing them, that kind of thing.

**Emily Lewis:** Since Lea and I are now working together, we're moving in this direction ourselves.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** I've been writing HTML and CSS as long as I can even remember. It's been forever, but I've never created starter files or base resources.

**Allison Wagner:** Really?

**Emily Lewis:** Though I would often basically copy what I had one from my last project.

**Allison Wagner:** Yeah.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** And then I tweak that, so in a way I sort of did, but with the goal of trying to be more modular. It's kind of as you described. I'm also inspired a bit by Jonathan Snook's *Scalable and Modular CSS*, then I've been trying to create these base resources and creating modules within them, and in my head, I was like, "Oh, I'll just create them and they'll be done." And no, I create them and every time I touch HTML, I find some new way I could approach it, so I go back and have to update those resources so they reflect the current knowledge, and then making them available to Lea so she's got those at her ready if she ever needs to utilize them.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** It's one of those things that it makes so much sense.

**Allison Wagner:** Yeah.

**Emily Lewis:** But you may not think of doing it because it takes time to get those resources together.

**Allison Wagner:** It does, but it saves so much time in the end.

**Emily Lewis:** Yeah.

**Allison Wagner:** And just having those class conventions that you already use, like we use `.btn` whatever as our button class, that make sense, right?

**Emily Lewis:** [Agrees]

**Allison Wagner:** And then `.more` for a read more link and that kind of thing, but just having those thing readily available and kind of like a second language that you just know when you look at a link, that that's what it should be classed and then everybody on the team also knows that that's what it's going to be classed because that's what we're using and that's our convention, it really helps speed



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

up the process to you because you're reducing your cognitive load of "I don't have to think about what I'm going to class it necessarily."

**Emily Lewis:** Right.

**Lea Alcantara:** [Agrees]

**Allison Wagner:** Or I'm just reducing that time, so it's just faster that way, and having those base styles associated with those classes, yeah, it's just a big time saver.

**Lea Alcantara:** Is there some...

**Allison Wagner:** I'm a huge starter files fan. [Laughs]

**Lea Alcantara:** So is there like some of documentation regarding this? Because, for example, you mentioned that, okay, if you see `.btn`, that means that's a button class that's going to be starting, and then there's another one that's called `.more`, but how do you decide which class would be abbreviated like "btn" versus spelled out completely like "more"?

**Allison Wagner:** I'm not entirely sure why we named things exactly how they do, but they make sense to me and I know them.

**Lea Alcantara:** Yeah.

**Timestamp:** 00:29:50

**Allison Wagner:** So we just continue down that path. I mean, I guess, we could get rid of all vowels or something like that, and just keep them as short as possible, but then there's that balance between like can I actually just look at it and know exactly what it is or do I have to look at it and then translate it in my head to English so that it make sense to me.

**Lea Alcantara:** [Agrees]

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** So kind of just finding that healthy balance and making sure that you actually understand what it is. I want to be able to look at a class and know what it does, and I think that also just comes with time and being familiar with the starter files. But once you're there, you're there and it's super helpful.

**Lea Alcantara:** All right. So before we go on, we'd like to take a moment again to thank our sponsors, Squarespace. Back when I taught Interface Design at MacEwan, a lot of students actually signed for Squarespace to have an attractive portfolio up and running really quickly, and that included a few technologically challenged students. So if you want something that easy, why not go for a free trial and 10% off your first purchase. All you need to do is go to [squarespace.com/click](https://squarespace.com/click) and use offer code CLICK. Now, back to the topic at hand.

**Emily Lewis:** I was thinking, Allison, when you were describing how you choose "btn" versus spelling out "button". I think one of the things that I've come to understand working just as a team of one but then also with Lea is that it's really picking something and going with it consistently, you know?

**Lea Alcantara:** [Agrees]

**Emily Lewis:** And I feel like it's one of those things where once you understand what the `.btn` class does, it's not really a matter of needing things to be overly verbose, but it just needs to be understood within the team, but not just the styles that the button has, but the styles that that class has on maybe a `<ul>` versus a `<p>` versus any other type of markup you may be associating it with. I feel like that's one of my biggest frustrations about some of the frameworks that are out there is that their class naming conventions, like people say it's so great for rapid prototyping, but if you have to spend ten minutes trying to understand their naming conventions, how is that useful? [Laughs]



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** Yeah, I mean, I think, well, the same would go for someone new coming into Happy Cog and looking at our starter files and having to understand them, but I think where the difference between the prototyping classes which are largely non-semantic and just speaking to sort of grids and what the actual appearance is and smaller texts and large texts, that kind of thing, it goes to the kinds of classes that they use. We're using classes that maybe are a little bit more semantic and relevant to what the actual content is inside of it. Because like I said, we do a lot of content-heavy sites, so there's going to be some repetition across. With what we're actually building, there's going to be callouts on the side as far as other types of contents. I see it time and time again, there are patterns that you can just kind of see from our various clients and making sure that we're coding for them appropriately.

**Emily Lewis:** So I know you wrote about frameworks for us a little bit last year for Web Standards Sherpa. If you have starter files, would you use those for a quick prototype, or would you rely on a framework, like an external framework or something?

**Allison Wagner:** For a prototype?

**Emily Lewis:** Yeah, like one of the things I'm finding as I'm building these base resources is that because my class naming is semantic and I understand it and Lea understands it, like I'd rather use my base resources to quickly throw something together rather than using something like Foundation or Bootstrap or something like that.

**Allison Wagner:** Yeah. So at Happy Cog, we actually have someone who kind of specializes in prototyping and he uses Foundation, and I have popped into help him at times and generally he's far enough along that I can kind of just look at what he's already done, and I'll go to the Foundation documentation as well and refer to that. But by and large, I can just kind of look at his templates so



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

far and just grab things and reuse what's already there as opposed to spending a ton of time trying to actually learn the system.

I guess it depends on how robust your starter files are. We have a grid system already kind of set up. It's in Sass, that kind of thing, so that it's already using functions to push out flexible width percentage based or whatever and using that. I'm not sure exactly what I would do actually if I was going to prototype. I don't know if our starter files are robust enough because our prototyping is super fast. I mean, he's pumping out a design a day kind of a thing, which is not the pace that I'm working on when I'm doing templates.

**Emily Lewis:** [Agrees]

**Allison Wagner:** So I don't mind Foundation. I think it's a good system, but I probably would be just more comfortable in our starter files because I'm so familiar with them, so I probably would take the starter files approach actually if I was tasked to do a full prototype.

**Emily Lewis:** Yeah, yeah. I think based on your answer, I think the question is, I guess, it's about your role. If your job is prototyping, which I didn't realize there was a job like that where you just cranked out wireframes and stuff like that, that sounds pretty cool, versus someone who's doing really custom front-end development. I feel like the latter myself or you might be more comfortable working with something that's like the starter files versus a framework. But if you're doing what your colleague is doing, yeah, I could see how that makes more sense. So I guess the question is a bit more nuanced.

**Allison Wagner:** Yeah, I almost feel like I would go into a previous internal project and scrape out anything that was highly customized.

**Lea Alcantara:** [Agrees]

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** And then just use that as my basis for the prototype of. I think I may be going that route.

**Emily Lewis:** So you've talked about some things that really sound like best practices in terms of markup and, by extension, templating. Are there any other best practices that you have in terms of workflows or even points of communication with your team members when you're doing templating for a content management system?

**Allison Wagner:** Well, we have a design-to-development transition meeting where we'll go through what's been designed thus far and kind of talk about values and identifying where things could probably be consolidated in terms of design to reduce dev time, that kind of thing. So I think communication and having a really good relationship with your designer is probably key, but then also, I'm a huge fan of SCSS, specifically, I'm using that, and I feel like that technology is at this point a necessity when you're doing large scale template dev. It just saves so much time using variables, that kind of thing, is crucial.

**Emily Lewis:** [Agrees]

**Allison Wagner:** And at this point, I mean, it's built into our starter files. That's how we do things and we're educating clients on SASS and SCSS and how to generate the screen file, that kind of a deal, and so far we've had a 100% buy in. Everybody loves it, so that would be probably the key is to make sure that your on the up and up with preprocessing. It's the way that we're moving and it just makes so much sense. It really reinvigorated my passion. [Laughs]

**Emily Lewis:** Totally.

**Lea Alcantara:** [Laughs]

**Emily Lewis:** I know exactly what you mean.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** It was another way to refine and abstract and make things better, smaller and smarter, and yeah, it forced me to rethink how I was templating and in writing classes and that kind of thing, it just saves so much time, and now when I have to go back and look at vanilla CSS projects and make updates on them, I'm like pulling my hair out like, "I can't believe I was doing this before." There is just like so many real things that can have helped save you time. Consistency is key, so just establishing those values and variables early, that will translate it across your full design system. It helps reinforce the idea of it being an actual system and not just like a bunch of styles that are attached to classes and that's attached to markup.

**Emily Lewis:** Right.

**Lea Alcantara:** So we've been talking a lot about your workflow with development and with other devs and all this kind of things, and then you kind of hinted at it just a few moments ago talking about client buy in. Now, clients are an interesting breed ... [Laughs]

**Emily Lewis:** [Laughs]

**Allison Wagner:** [Laughs]

**Lea Alcantara:** And that sometimes they have a lot of technical knowledge or not very much, and certain clients want a lot of customization and other clients are just fine in putting content into the CMS. I'm wondering if you believe that the clients need to access certain sections or customization options like layout options and things like that. Would you build a template differently for that CMS to allow access, for example, for a client to certain parts of the template, or do you lock that down?

**Allison Wagner:** Yeah, I mean, I want to make sure that the template system that they're handed is flexible enough to certainly handle all of the content that they're going to need to support on the site,



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

but then also there are layout varieties, that kind of a deal. Make sure that that's being accurately represented and that things don't fall apart if it's missing a sidebar or something like that.

**Lea Alcantara:** Yeah.

**Allison Wagner:** So we want to code for all of those instances and just kind of making sure that the outliers are being taken care of. By and large, the companies that Happy Cog is working with are large and have someone who is fairly proficient, at least, in web on their team internally to kind of take things over.

**Lea Alcantara:** [Agrees]

**Allison Wagner:** We don't do a lot of maintenance per se, at least right now, I think we're maybe moving into that, but by and large, we would come in, do their website and then kind of hand off to them to continue to build because the website is never really complete right. You're always adding content to it, always changing things, you know?

**Lea Alcantara:** [Agrees]

**Allison Wagner:** There are new technologies, that kind of deal, you have to keep it evolving. So making sure that the client is very comfortable in the CMS and able to add content and is not scared of the system. It's all about education, just making sure that they're on board with the CMS chosen and that they feel a 100% confident that the system that we built for them is going to take them into the future and they're going to have many years of success. I mean, I don't know how many years of success, but at least several hopefully, you know?

**Timestamp:** 00:00:40:05

**Emily Lewis:** Yeah.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** I would say that's a sign of success.

**Emily Lewis:** I think, at least thus far with my, I guess, freelance career, most of my clients lack understanding of HTML and CSS, and so for the most part, I do not let them touch templates. If they do, they're going to end up paying me a lot to fix what they messed up.

**Allison Wagner:** Yeah.

**Emily Lewis:** And so one of the things I try to do is identify what types of content they need to enter and where can I avoid giving them a WYSIWYG where they could actually affect HTML.

**Allison Wagner:** Yeah.

**Emily Lewis:** Where can it be just discreet content that they don't have the ability to mess with the markup, and then where they do get a WYSIWYG editor where they could potentially go into the source and enter HTML that I've tried to restrict based on maybe the formatting dropdown or the styles dropdown in the WYSIWYG editor, depending on which one you're using, but wherever that content falls inside an HTML template, I wrap it in a `<div>` called `.from-editor`.

**Allison Wagner:** There you go.

**Emily Lewis:** And then I overwrite all of this. I'll just imagine what they could possibly enter into them.

**Lea Alcantara:** [Laughs]

**Emily Lewis:** And I use that to kick off the string for the selector to make sure that all the styles, even if they put an `<h1>` in there when that's not technically what we want them to do, it's going to visually look at least the way it should as it were maybe an `<h3>` which is what we did want them to have.



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** I feel like that would be a really fantastic code snippet. It's like all the worst client content entry that's available.

**Lea Alcantara:** [Laughs]

**Emily Lewis:** [Laughs]

**Allison Wagner:** And then have that in your template over just to test against to make sure that when they're entering content into the WYSIWYG, that it does print out correctly, because that's what I do too. I will add a class of WYSIWYG or something to a field that's going to get a WYSIWYG and test to make sure that when things don't line up correctly or when the markup that was, I guess, generated by WYSIWYG isn't a 100% awesome or less than, you know?

**Emily Lewis:** Right. Or throws in a bunch of line-breaks every time or things like that. [Laughs]

**Allison Wagner:** Yeah, `<br>`, `<br>`, `<br>`, `<br>`, `<br>`, `<br>`.

**Emily Lewis:** [Laughs]

**Allison Wagner:** Yeah, exactly. Yeah, with that kind of stuff, we're making sure that things don't fall apart, but then it does kind of come to them as well. They need to be looking at the previews or the published pages and making sure that the content is entered or is printing correctly.

**Emily Lewis:** [Agrees]

**Allison Wagner:** And then also I'll go in after a content is entered and kind of clean things up and make sure that things look okay and then let them know what kind of things moving forward they should watch for, that kind of thing. So it's kind of like a lessons at the end.

**Emily Lewis:** Oh.

**Allison Wagner:** Yeah, to help to educate them.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** Oh, that's a nice idea. I like that.

**Allison Wagner:** Yeah, just so that they don't make those mistakes moving forward after we're not actively engaged with them. Yeah, they appreciate that, and then they will also post bugs in Sifter or that kind of thing if they're having issues with a content entry where I can just see it right away, but yeah, that's a matter of the front-end dev coming back in after back end and content is entered and making sure that things kind of look appropriate.

**Emily Lewis:** Oh, I like that stuff. Lea, I think we need to get that in our workflow.

**Lea Alcantara:** Yeah. Now, I mean, that reminds me a little bit, even in the design world, when you're doing identity design and in the traditional print world, whenever you design a logo in the brand guidelines, you have an entire page saying, "This is what you should not do. You should not angle the logo this way. You should not stretch the logo this way." And in some ways, basically in the WYSIWYG, it's like, "Okay, here are the best practices and here's what you should try to avoid so it doesn't break."

**Allison Wagner:** Yeah.

**Lea Alcantara:** Yeah. Pretty cool.

**Emily Lewis:** So I kind of wanted to, for the last bit of this conversation, talk about the notion of generated markup, and by that, I'm talking about whether it's the CMS or even like an add-on for the CMS, and what it spits out is not just the data. It's also whatever the CMS or add-on developer deemed was the proper markup for the given output. For example, I remember, I think this was before EE Version 2, the comment form.

**Lea Alcantara:** [Agrees]

---



<http://ctrlclickcast.com/episodes/cms-markup-templating>

Transcript sponsored by



---

**Emily Lewis:** You just put in a tag and it generated all the form markup for you, and I couldn't get in there and make my labels and inputs and all that stuff, what I semantically wanted it to be. So with that, what do you think about generated markup with a CMS? I mean, is it something that you have had to work with, and if so, what kind of challenges did you face?

**Allison Wagner:** Yeah, so I think that starts with choosing the right CMS that has a really robust developer community and has strong plugins because not every CMS is just going to have out of the box exactly what you need based on the content or the client requirements, that kind of thing. So doing your research ahead of time and choosing a platform that has really nice plugins, but then again, doing research before you select the plugin to make sure that the markup that is generated is semantic at the very least. I guess that's what you're hoping for, and there are always going to be times where you're unfortunately doing things with templates where you're just kind of like, "Well, this has to be done." So if you have to wrap that form in a `<div>` and have a class on it and then target it that way, then so be it.

There are checks and balances kind of a thing. So while it might be less than ideal, it is kind of for the greater good as you're moving the project forward. Those kinds of things happen. I haven't run into them too much with ExpressionEngine thankfully. I know Drupal generated some markup that was less than ideal that we had to kind of work around, but then it's just like going in with the CSS and kind of cleaning it up where possible. It's what it is. It's less than ideal to begin with, but I think that there are appropriate solutions for it to make it at least print appropriately and then worrying about accessibility after the fact, you have to go in and if you can clean it up. I mean, that's also reaching out to the developer of the plugin if possible and asking for advice on how to tackle this problem, that kind of thing. So just making sure that you're asking the right questions and if you need to make updates, then so be it.



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** Yeah. I encountered this issue myself recently for a project we're just finishing up this week. There was no other way, but I was pissed off by the whole thing. [Laughs]

**Lea Alcantara:** [Laughs]

**Emily Lewis:** So basically, we decided to use Structure for an ExpressionEngine site to manage that. Primarily, our goal was to have it managed the navigation of the site, but the markup that Structure spits out for, let's say, for example, the primary navigation on a site, it's limited and there isn't a whole lot you can do to modify it, and this was a responsive site so there is a need for a few mobile toggles in place and a need for some special classes on the child.

**Lea Alcantara:** Yeah.

**Emily Lewis:** There were secondary dropdowns and so my template, the markup was beautiful, and then I dropped Structure and I was like, "Oh crap." I can't actually do what I need to do especially from a responsive standpoint with the markup that it generated and there wasn't a lot of opportunity to customize it.

**Allison Wagner:** Oh yeah.

**Emily Lewis:** And so I found a plugin for a plugin.

**Lea Alcantara:** [Laughs]

**Allison Wagner:** [Laughs] Oh yes.

**Emily Lewis:** Oh, what was it called? Structure... oh, I'm blanking on this.

**Lea Alcantara:** Entries. StructureEntries.

**Emily Lewis:** StructureEntries.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Lea Alcantara:** I think so.

**Emily Lewis:** And that lets me generate my own kind of nested UL structure, but it was a hog in terms of the query resources. I mean, the page was like eight seconds to load after it was done parsing.

**Allison Wagner:** Oh.

**Emily Lewis:** And thank God, there are caching tools. We used CE Cache from Aaron Waldon, and it kind of resolved the issues I was having, but I spent way too much time trying to get that to work to the point where, in my mind, I can't think of a reason to use Structure again, especially if we're going to be doing responsive builds where I do need more customization in my navigation.

**Allison Wagner:** Yeah, I wonder how else you would approach that. In my head I was thinking, "What if you take JavaScript and you rewrite it." But that's like stupid. No, no, no, no.

**Emily Lewis:** [Laughs] It's one of those situations where you can always find a way to work around it if you have enough time, you're just going to get it done. You're convinced yourself.

**Allison Wagner:** With tenacity, yeah, yeah, yeah.

**Emily Lewis:** Yes, but it's kind of like, was it worth it? I kind of wish I had never chosen that add-on in the first place. I feel like I could have done something differently even with native ExpressionEngine right after the box, that would have been faster than what I had to deal with, what I thought was going to make my life easier.

**Allison Wagner:** Oh yeah, and because once you go the Structure route, you're kind of have pigeonholed.

**Emily Lewis:** You're in it.

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** Yeah. You're not going to...

**Emily Lewis:** Yeah. [Laughs]

**Allison Wagner:** Yeah, you just have to restructure the entire CMS. Yeah, I know. Yeah, that's a tough one.

**Emily Lewis:** Yeah, it was a good lesson learned in the sense of it's going to make me a lot more careful about what we recommend for that particular functionality in ExpressionEngine moving forward, and I'm going to be a lot more focused on paying attention to, regardless of what the CMS is, what that markup is, just to make sure that I'm not digging a hole for myself that's going to take me forever to get out of.

**Allison Wagner:** Yeah, that's actually surprising to me because, I mean, I've used Structure before. We had used Structure in client projects and it's very well received.

**Emily Lewis:** Up until now, I have used it and loved it.

**Allison Wagner:** Yeah.

**Emily Lewis:** And I still love it.

**Allison Wagner:** Yeah.

**Emily Lewis:** It's just that for that need to generate nested ULs...

**Lea Alcantara:** For responsive...

**Emily Lewis:** Yeah.

**Lea Alcantara:** Yeah.



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** I needed to throw toggles in there and I needed to do different things that I couldn't do with my skill set in that plugin.

**Lea Alcantara:** [Agrees]

**Emily Lewis:** So it was crazy. [Laughs]

**Allison Wagner:** [Laughs]

**Lea Alcantara:** But speaking as maybe a devil's advocate, is there ever any situation really where generated markup is necessary? Is it good?

**Emily Lewis:** I wish it wasn't. I wish it wasn't. I feel like people who build add-ons have their skill set and it may not be markup.

**Timestamp:** 00:50:01

**Allison Wagner:** Yeah.

**Emily Lewis:** Just output the data, let me do what I'm good at. Let me have the flexibility to control what the markup should be for a site because that's what I'm good at. That's my opinion. [Laughs]

**Allison Wagner:** Mine too.

**Lea Alcantara:** Allison, how about you?

**Allison Wagner:** Yeah, now I agree with you entirely. I guess it's not a perfect world. There is always limitations, that kind of thing, so I think also now we're in creative problem solving. That's part of our job.

**Lea Alcantara:** Yeah.



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** So finding the solutions to those kinds of problems, that kind of makes my day a little more exciting. Well, I don't wish them on anyone. Sometimes, working around those problems that are going to come up with all the third-parties that we work with as developers, there's going to be snags and that kind of a deal and finding the most appropriate solution is a challenge, but I mean, they're not the best, but they're not...

**Emily Lewis:** They're the ones you learn the most from.

**Allison Wagner:** It's fun to solve problems.

**Emily Lewis:** I think that's the stuff you learn the most from when you really have to deal with something like that.

**Lea Alcantara:** Yeah, yeah.

**Allison Wagner:** And do like a deep dive into something that you're not familiar with and just kind of being forced to do it because there's no one.

**Emily Lewis:** [Agrees]

**Allison Wagner:** Yeah, that's it. I do well when I'm backed in a corner and I'm like, "There's no other options and you just have to make it work and you have to learn this." And that's kind of how I thrive as far as my skill set goes, which has its plusses and minuses for sure, but when I'm forced to learn something because I have to, then I'll do it and I'll be a better dev for it.

**Lea Alcantara:** So speaking of helping yourself become a better dev, if you were speaking to a CMS developer right now and they're asking you how to improve your workflow, job and life, what would you ask them to do to make templating and manipulating markup easier for a CMS?



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Allison Wagner:** I don't know. I feel like it's more my job to help them with their job than it is for them to help me with my job. But I think just having a really sound collaborative kind of partnership and making sure that we're communicating and bringing projects or problems rather up early before they become a big problem is kind of key to just having open communication. I start from starter files and my constraints I feel are far less than what they're dealing with working in a CMS and having to deal with all the CMS overhead.

So I kind of have the attitude that I'm trying to make their lives easier, which maybe in turn they have the attitude that they want to make my life easier. But I don't know, I just feel like making sure that both people are kind of working together to make sure that the end product is what it should be in where it needs to be really gets the job done.

**Emily Lewis:** Now, before we finish up the episode today. Do you have a final pieces of advice for those of us who are templating for a CMS?

**Allison Wagner:** Yeah, just really focus on when you're writing your CSS, do yourself a favor and look for the common patterns across the design system and write those first and then overwrite carefully when necessary, but if you have your sound base to begin with and all your low level or I guess high level CSS written early, you'll reduce a ton of time in the long run.

Yeah, and then also just having those starter files, but just making sure that you're maintaining them, updating them and very comfortable with them so when you look at the design, you just kind of already know what it should be classed and what kind of style should go along with that, and then where it needs to be overwritten. I guess also, avoiding the body classes as well, accepting a certain very specific instances because they're so high level and they can become troublesome quickly, and just using those sparingly and keep away from importing, that kind of deal.



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Lea Alcantara:** Learn preprocessors.

**Emily Lewis:** And just the basic good practices.

**Allison Wagner:** Learn SCSS. [Laughs]

**Emily Lewis:** Yeah.

**Allison Wagner:** That's key. That would be my biggest one, learn that. [Laughs]

**Emily Lewis:** Yeah, it sounds like just really good foundational workflows and naming conventions and approaches.

**Allison Wagner:** Yes.

**Lea Alcantara:** Perfect. So that's all the time we have for today. Thanks for joining us, Allison.

**Allison Wagner:** Oh, absolutely. Thank you so much for having me. This was awesome.

**Emily Lewis:** We loved it. In case our listeners want to follow up with you, where can they find you online?

**Allison Wagner:** My Twitter handle is [@alliwagner](#) and you can shoot me an email, [wagner@happycog.com](mailto:wagner@happycog.com). Yeah, that's about it.

**Lea Alcantara:** Awesome.

**Emily Lewis:** So thanks again.

**Allison Wagner:** Awesome, thank you.

**Lea Alcantara:** [Music starts] We'd now like to thank our sponsors for this podcast, [Squarespace](#) and [Pixel & Tonic](#).

---



Transcript sponsored by



<http://ctrlclickcast.com/episodes/cms-markup-templating>

---

**Emily Lewis:** We also want to thank our partners, [Arcustech](#), [Devot:ee](#) and [EE Insider](#).

**Lea Alcantara:** And thanks to our listeners for tuning in! If you want to know more about CTRL+CLICK, make sure you follow us on Twitter [@ctrlclickcast](#) or visit our website, [ctrlclickcast.com](#).

**Emily Lewis:** Don't forget to tune in to our next episode when we'll be talking with Ben Parizek about Craft plugins. Be sure to check out our schedule on our site, [ctrlclickcast.com/schedule](#) for more upcoming topics.

**Lea Alcantara:** This is Lea Alcantara ...

**Emily Lewis:** And Emily Lewis.

**Lea Alcantara:** Signing off for CTRL+CLICK CAST. See you next time!

**Emily Lewis:** Cheers!

[Music stops]

**Timestamp:** 00:55:03