2024-25 GAWD “Web” Advisory Board meeting minutes.

Faculty at meeting:

- **Will Bohmann**
- **Matt Cronin**

Web Advisory Board Members:

- **Dave Gibson** - Propeller Media Works
- **Matt Onyon** - Keene State College
- **Mark Dubois** - Web Professionals
- **Adam Fehnel** - Franklin Templeton
- **Matt Gonzalez** - Winooski School District
- **Chris Goodwin** - Zotec Partners
- **Steve Herr** - Rovers North
- **Tim Basiliere** - Franklin Templeton

2024-25 Update

1. **Rebranding**
2. **Workforce Development**
3. **Year 1 curriculum updates / less coding, more CMS**

1. Rebranding

In consultation with our Advisory Board we are happy to announce that our name has officially moved from Computer Animation and Web Page Design to **Game Animation and Web**

Development. We hope the updated branding, losing some legacy words (Web Page), and adding in some of our newest curriculum (Game Development) will be more relevant to our customers (high school students).

2. Workforce Development

This year I would really like to hear input on Workforce Development. The debt and time cost of college precludes many of our students from even attempting to matriculate, while these students may be the best and brightest in our group.

I want to find ways to get students from high school to gainful employment without the costs of college, and I would like your help brainstorming ideas.

I reached out to the Vermont Department of Education to ask about some sort of licensing program, even if just to verify that these students are in fact hard working, good communicators, and timely. Ready for an entry level position. The Vermont DOE responded back with “it’s on the employers to request this” industry wide, so that wasn’t very helpful.

My next goal is to reach out to big employers in the Chittenden County area and see what their workforce development goals are, and see if we can create some sort of a pilot program where the best and brightest of our program (top 2-5%) could get at least interviews for some of the entry level positions at the end of their year with us. I am going to pull on any heartstring I can about Vermont youth, saving money in post secondary training, employment, etc.

We have lots of hard working youth in Vermont that are great with technology, communicate, are hard workers, and can learn on the job. How can we help them? I can't imagine the best option we have for these young adults that can't afford college is freelance, YouTube University, and pull yourself up by the bootstraps. There have to be other ways to tech jobs that don't require college.

Focus 1) With the cost of college and AI on the horizon, I am looking for ways that hard working students who communicate well and hit deadlines can find gainful employment. What are the ways that new employees to the industry are finding employment? I am trying to find tracks that aren't just the typical "get a 4 year degree".

If you are an employer or work at a business, are you hiring right now? What are the roles your business is actively looking for? What are the background requirements? Is a degree required for the roles you are looking for? If so, why is the degree required?

If you are in education, what types of jobs are your students getting, and what type of education opened the door to these jobs? I care less about Jenny that went to MIT and got that app development job, I want to know about Josie that went a more affordable route and found success.

> **Dave Gibson -** I've had a small but stable team, so I haven't had to hire in about 6 years, and it's been 12 since I hired anyone that codes. So I'm not all that knowledgeable.
>
> You are definitely wise to get away from teaching coding. AI will do that. And while we do often code Wordpress or Shopify themes from scratch, that time is about to end

completely. Since we're so focused on accessibility, we have created our own themes that have as much baked in as possible... but again, that's about to end.

I see only a very limited future for anyone coding. I do still see a future for UX designers, but again with heavy AI use. The designer who owns the UX and functional vision for how things work and are used, will now have the AI assistant to do coding/implementation.

Mark Dubois - How are certifications being factored into this equation and how are they being evaluated? For example, many certifications come from testing companies which only teach to the test. As a professional organization, we don't see the value in those. We also see limited value in tool certifications. If you know how to use a tool today, the tool will likely change significantly in the next year (as more and more AI is incorporated)**.**

Chris Goodwin - I have worked with plenty of smart and knowledgeable folks who didn't formally study web development. I think there is certainly a pathway there. However, as with anything, there are plenty of tradeoffs to consider.

I have found that many of the places I've worked or applied to do require a degree. They don't always care if that degree is related to the work being done but they do require one. Of course, you and I both know this is ridiculous. I believe many employers see a degree as a stamp of approval that a person is able to function in a work-like environment. Further, they believe there is proof of an ability to follow through and show up.

I certainly do not know but I believe it may be difficult for folks to break into an already competitive market without a degree. That said, I also believe the way you're thinking about it is correct. Focusing on hardworking, good communicators, who are timely is key.

I think the most likely path forward is to try and leverage folks who have ties to GAWD already and see if there can be any kind of program in place to help bring on developers in JR roles consistently. Of course, there is also an opportunity for you or the school to reach out to places and try and get programs set up.

We are hiring at my new gig (Zotec Partners) for a few developers. We are advertising for Senior roles only right now. I think this trend is quite broad right now. There are some companies with developer needs but many of them are looking to spend more and train less. This type of approach is not an issue individually but when the entire industry is leaning in that direction it will lead to a lack of quality developers growing into their roles and becoming the next Seniors. Time will tell how this sorts itself out.

Random note: I have been constantly surprised by who in my network has led to my next job. When I started at DockYard (my second position out of college) it was because someone I went to college with reached out and told me to apply. He and I were friendly in school but we weren't close. When I was unemployed after a round of layoffs, someone I had only worked with slightly at DockYard reached out with a work opportunity. He knew someone from local developer meetups which was looking to hire. That person happened to be the head of engineering at Zotec.

My point here is twofold:

1. You build trust and earn trust through your connections. People see how you work and interact even if they aren't particularly close to you. You can't fake this. You either show up every day willing to do it or you don't. People will notice.
2. The network is everything. People always told me "networking networking networking" but they never actually explained what it meant. It means building authentic connections with people. It's not handing out business cards, it's going to meetups and speaking, working on passion projects, and finding your people.

GAWD certainly has its own version of these networks, maybe it's time to start trying to build those bridges between school and employment. (it's also possible y'all are already doing that, if that's the case, what is working? What isn't working? I am happy to help brainstorm more about this too)

Matt Gonzalez - I am watching students get jobs and internships for media production in order to get into the door at a tech company. Right now we have a former student at New Breed for an internship for media marketing and video production. Part of that is creating video for the web and displaying it on his own and company website. If you were to develop a survey for me to share with contacts at local companies, I would be happy to help collect responses from them.

We also do have a former student that went through Vermont Tech to get a degree in software engineering and is working in the field as a full stack "software engineer" (web developer). I am seeing a lot a vails being placed over "Web Development" to sound more specialized or sophisticated.

> **Tim Basiliere** - This seems pretty brutal in the current job market. All junior listings I’ve seen require college degrees. I’m not sure what the best approach will be in the future given the emphasis on seniors with degrees in the job market. Portfolios may help, but AI filtering is a problem as well.
>
> That said, the only way I can imagine success that early is networking. Maybe getting your foot in the door with larger tech companies in the area that see the value of long term employment who will invest in teaching via internship. It would be nice if even SkillsUSA had an employer outreach program to connect employers with excellent performing students.

Adam Fehnel - I know a lot of young adults are doing coding boot camps, either alone or in tandem with a degree. It's been a few years since I've been involved in the interview/hiring process, but we saw lots of applicants with these 8-10 week programs with rather decent portfolios based on the work and learning they did there. That said, there was a cookie-cutter feel to these applicants and few of them actually stood out from the masses. I'm sure just like all things, there are some good ones and a lot of copycats just doing it for the money. I don't have any good examples, but I'm sure with proper research we could find them.

Franklin Templeton is in a huge staffing reduction period right now. They've been acquiring companies like Putnam and merging the workforces, which has left them with a ton of bloat. I worry my name might come up eventually, especially with the market turmoil and being in a financial company. We are definitely not hiring.

3. Year 1 curriculum updates / less coding, more CMS

As the web continues to evolve, I think we have hit that tipping point where any work done has to have a client side CMS.

In our year 1 program we are embracing this reality by adjusting our curriculum. No longer will we spend as much time on hand coded layouts and programming, instead using that time to deploy and customize CMS powered designs, specifically using Wordpress as our CMS.

Our year 1 Website Development coursework (3 college credit earning through the Vermont State College System / CIS 1151) coursework will follow this flow:

- **Quarter 3**
    - Hand coding HTML, CSS
- **Quarter 4**
    - CMS Installation, customization
    - Hand coding of JS / SQL database intro

Focus 2) How do members feel about this? I would LOVE to spend an entire year going through the basics of code, and I see the academic value of it, but as a technical center trying to lean in to practical projects, we have to move to CMS development, Wordpress specifically, as that seems to be what many students will encounter in the wilds of 2025. Our students could start doing freelance work this summer - and more than likely it’s going to need a CMS.

> **Dave Gibson -** From a platform standpoint, different platforms are appropriate for different companies and functions - primarily between ecom and non-ecom, and then companies with or without a web team due primarily to size. Mom and Pa shops are

better off using Square or Wix that offloads the entire hosting/backend/security burden and liability to the provider for example.

Marketing websites: Wordpress. No question.
Ecommerce: Shopify.
Mom & Pa: basic ecom/basic brand site : Wix.

Also, digital accessibility is about to get more attention. The European Accessibility Act kicks in this June. The EAA will apply to any US business that touches EU citizens. Has potential to be the next GDPR. Here's my blogpost on that.

Then ADA Title ii, which applies to state to local government and entities funded from such - from local library, to .edu to .gov. :Like Sect 508, which got updated a few years ago, they call on pretty much anything paid for with public funds must be accessible.

And this applies not just to web design - but actually more so for SaaS.

Honestly, I would heavily concentrate on teaching how to use AI ... which means learning how to envision, plan and how to articulate what you want to accomplish - and then how to test to verify - which requires understanding of programming principles and best practices for quality assurance verification.

Programmers should make sure to learn principles of design and how to administer the CMS... how to assemble sites, not build them.

I wish the future looked brighter for these careers, but without question, AI is going to massively disrupt this area.

Mark Dubois - Regarding the hand coding - it is probably worth recognizing Vibe Coding and planning on how best to incorporate that into the curriculum (perhaps even modifying Vibe Coded examples).

Wordpress powers roughly 40% of the WWW these days. I think you are moving in the right direction. You might also consider having more advanced students migrate from one CMS to another (Joomla or Drupal to Wordpress, for example). It is not as easy as one may think and this is a niche I don't see a lot of schools embracing. I would recommend not only understanding the basics of a CMS, but also how to customize it. For example, students should know how to create a child theme in Wordpress and modify that. Perhaps they could also write a simple plugin to understand how to enhance the CMS. I believe that would give them extra value as they start their careers.

Chris Goodwin - I may be an exception to the rule, I'm very curious about what others say to this question.

My experience has been that expertise and understanding of *how* things work and *how* to write code from scratch is exactly what folks are looking for. They want developers who understand how to get things done and know the fundamentals of how to write code. Every job but my first has been a role specifically dedicated to expert knowledge about how to build things from the ground up, from scratch, with hand written code.

In my experience, companies are seeking that kind of expert knowledge.

Also, Q4 "hand coding of JS" - yes please! Somehow I did 4 years of college with ZERO classes on javascript. Had to learn it all myself. ABSOLUTELY RIDICULOUS.

That all said, I can see a whole other role. A role more focused on quicker building for smaller sized clients/companies who need a web presence. Those projects lend themselves well to wordpress, shopify, etc. I know folks who have had success working in those roles. While it often offers more freedom and a more flexible schedule, it usually comes with less pay, a requirement to be your own boss, and admin work. It's a viable route but for a certain type of person.

Random note: If I wish I'd done any one thing while I was in college, I wish I'd learned how to write shopify templates. I wish I'd had a whole class on it. If it were easy for me to spin up a few templates and iterate them over time, I think I could have multiple themes providing passive income (no income is truly passive, you'd have to maintain the themes).

If GAWD is interested in helping empower students and teach them how to work with CMS's. Focusing on one which they can build as part of an ongoing project and potentially release to the world could be a great way to help the growth begin. Give them the knowledge of how to build a template up to the required standards to sell it, and you've helped them find a way to employ themselves. I don't know if its a cash cow but there is certainly money there.

Matt Gonzalez - I am in agreement with you. I feel the need to get basics understood first before making it very complicated with involving a CMS. It might be a per-student basis but I really feel like it can be overwhelming for students to try to troubleshoot if something is a coding issue or a CMS issue. I am not sure what your curriculum looks like now, but developing hard-code first and understanding it before plugging it into the CMS would be ideal. Even if it was one module or section at a time would help understand the coding while getting use to navigating the CMS. If you're trying to get your students to learn a CMS for them to freelance, it's possible to teach them as a CMS user first to update content and then when they are familiar with it, learn to create code, create a template, and activate it. I fear a lack of understanding could land them in a tough situation if they take down a website.

> **Tim Basiliere -** I suppose this depends on the students and what their individual goals are. Using a CMS to create sites is more web production than web development, but maybe development is best suited for year twos. Why not use headless Wordpress CMS and have them utilize the REST API? This way they're familiar with one CMS front end, but also familiar with an output every CMS will provide. This is extremely common, and using fetch requests to use/manipulate API data is required. Maybe less focus on basic HTML/CSS and revisit later after CMS is set up?
>
> I'd suspect people hiring will want someone who can accomplish a lot more quickly in a stateful framework, and knows how to debug the code. 10 minute vibe code (sorry) sessions and teach how to debug? React is also pretty much a necessity if the goal is to get a dev job out of the gate.
>
> Coding may be going away for anything other than a masters degree in the future, but it seems like it’ll be a gradual exodus over the next couple decades. Knowing basics and structure is still extremely important, and as more people rely heavily on AI, people like

> me will become rare. AI is horrific at debugging at the moment, although GPT 4.1 came out and I'll be trying that out this week. Maybe another baby step.
>
> I think it’s become a bit of both technical and basic for web dev, at least. Technical in that using non-technical tools demands a strong understanding of what’s happening. It’s definitely a bit more forgiving for those who are less technical, but it can snowball quickly. That said, as testing and debugging AI comes around, maybe it could. That may be a few years, though. Agreed, I hate all of this uncertainty. All I really want to do is make fun user experiences. I couldn’t care less about coding, hah.
>
> Maybe this is an opportunity to lean into heavy AI pair programming. Even AI itself for students interested? It must be rough to build a curriculum on something that changes so quickly, though.
>
> Right now interviews are mostly leetcode and live coding practices. I’ve seen live coding more. Simply getting interviews are rare, and most of my success has been through networking. So from personal experience and from the experience of what I see in my network, soft skills and networking seem most important. I'm interested in how other folks feel about this, since it takes so long to develop coding expertise.

Adam Fehnel - WordPress is still ubiquitous and not a bad idea for a starter CMS. I know there are simpler solutions with less bloat, but the reality is that WP still makes up the vast majority of sites. I also appreciate hand coding sites, especially when first learning to understand the foundations of front-end and how the browser renders html, but there is value in adding more CMS to the curriculum.