

EPISODE 21**[INTRODUCTION]**

[0:00:00.9] NA: Welcome back to La Vie en Code, a podcast dedicated to the self-educated web developer. I'm your host, Nicole Archambault, and today we're going to be switching it up a little bit in talking about hackathons. A lot of folks think of hackathons and probably imagine some uber competitive, brutal weekend of elite hacking, whatever hacking they imagine to be. I mean, maybe some hackathons are like that, but I've only been to one so far and it, as well as the next one that I have coming up here in November, have both been women's hackathons.

So we'll talk a bit more in a bit about why they tend to be different. But generally speaking, know that hackathons are not these super intimidating competitions. In fact, more and more hackathons are aimed toward helping people who are new to the industry to gain confidence and brainstorming solutions and coding collaboratively.

I'm going to share my experience with my first hackathon, which I won, as part of the team of women at the We Code Women's Hackathon back in 2015. I'm going to extract what I learned from the process and the experience that has helped me in my career even two years later. I'll also give you some tips for how you can do a bit of preparation for your first hackathon so you're not completely unprepared. I'll give you a spoiler alert; like a lot of things that I tend to talk about on the podcast, it has nothing to do with actual programming. This isn't because programming isn't important or because I don't know how to program myself but rather, for a couple of reasons.

First, because there are tons of resources out there for the programming aspect, which a lot of people often don't even know how to leverage properly. Second, because for hackathons in particular, you can generally hack in whatever language you prefer. The programming itself is irrelevant. The preparation that you'll be doing instead for your hackathon is more in line with knowing how to develop and flesh out your ideas, how to contribute your ideas, and how to help assign roles to your team based on each person's individual strengths.

So remember, you have very little time to organize all of this. These hackathons are usually anywhere from like a day or two to like entire weekend. But you're developing an entire idea while socializing and doing like all the fun things that hackathons offer. That time ticks down super-fast. So we're going to jump right in to the good stuff and get you ready to tackle your first hackathon.

At the end of the episode, I also have a special freebie for the listeners. I put all the information from today, shortened down into a checklist for you to use to prepare for your first hackathon and hell, even to take with you to ensure that your focus is on the right stuff and really get the most out of the experience.

Okay, everybody ready? Let's go.

[EPISODE]

[0:03:36.6] NA: Let's start with what hackathons really are. Hackathons are effectively social brain storming sessions for problem-solving using tech. They are not programming marathons, where you do nothing but program for hours. They're not isolated events; they're designed to be highly social and fun. Hackathons help the world by bringing dreamers and thinkers together to develop ideas together. You can tell by my emphasis on together that there's something there and yes, there's something very big there.

There are stereotypes of the "web developer" being a loner and even if they're not true, stereotypes tend to affect the way that people see themselves as well as the way that others see them. Web developers are social creatures and we should be. How are you supposed to learn if not from the work, the writings, the recordings and the lessons of others? How can you connect with those people if you've never communicated with them?

One person can develop a single idea but a group of people can bring that idea to life and present it in a way that considers several different perspectives. It's incredibly important that you understand that aspect of hackathons because if you don't consider yourself to be like "the most social person", then you need to commit to improve in that area before you can consider yourself a good web developer.

Sure, you can be an introvert, I am a massive introvert. I can and often do spend entire days to myself like on end, working on my business and online course without seeing or talking to other people. However, eventually, you have to go out and build and maintain your relationships and share your ideas and your work. If you leave that part of the equation out or you think it's not important, then you'll be sorely disappointed to realize how many opportunities you'll miss out on in the web development industry. If you only consume and you never reach out to share your own ideas, you're missing out on a huge part of what makes tech amazing; the exchange of ideas between brilliant people.

I've participated in one hackathon to date, which I enrolled in when I was only 60 days into my web development education. I was far less confident as a web developer, as a professional, to generally as a person when I first considered whether or not I should register and go through with it. I wrote about the experience on the La Vie en Code blog called *Baby's First Hackathon* but long story short here, I have a lot of things that I would have done very differently had I known what to expect.

It's also important to note that this was a women's hackathon, which was an enormous part of why I felt so comfortable attending a hackathon in the first place. I didn't feel like I really have my legs under me and my head in a good enough place to be able to take programming feedback from someone if I felt they weren't being patronizing or unhelpful. Now, I have a much better sense of how to handle those situations and a lot more confidence in my newbiedom, in fact I embrace it.

But I went to the two day we code women's hackathon. Hosted by Nike and Puppet Labs in Portland, Oregon back in 2015. I knew nobody else there, which was pretty tough actually. But I pushed through all of my discomfort and pushed myself out of my comfort zone entirely. During the opening ceremonies. The organizers explained the challenge: to create a web or a mobile app that would address issues that women in technology and women in general commonly face.

I was assigned to a team of I think four other women of all ages and backgrounds. Their backgrounds included design, development, user experience, marketing. I mean, it was really kind of a Motley Crew. That's typical, as I've heard. I had no clue what my experience had given

me in terms of strengths at that point, but I still stuck my neck out and I ended up providing our winning idea. It was a stack overflow-like site. Bound by a very strong code of conduct and terms of use that would provide women a safe space for ask questions and get helpful answers.

We built on to the idea later to add a mentorship connections, but trust me, not all of it was in place when we present it, not at all. It's kind of funny because I was actually really worried about being able to fulfill the obligation of creating and fleshing out this thing that I had dreamt up as if it was like me alone that had to do it. But I had never worked on a team, developing an app before or developed an app at all for that matter, 60 days into my education, and I was really happy to see that it was actually working out.

I went home that night because I was super anxious and felt like, you know, I needed to be in my own bed, in my own environment and the second day, we began designing and programming our app. I had learned some Git from Treehouse but I totally solidified my knowledge by having to figure out commands on the fly in order to contribute my code and my files. We took a bit of an inventory initially to see where everyone was in terms of their skills and their strengths and I realized that we actually had a pretty solid team to handle the basics.

I worked with another team mate kind of fumbling around with some CSS, I took some time to observe other people and generally, from the second day, I had a blast from that point on. When the development portion was over, it was on to presentation time. We presented dead last and we were aptly named Lucky 13, we named ourselves that because we were the 13th group.

I've never believed in superstition though, so we just own it. We had divvied up the presentation itself so that we all had something to focus on and sell. Like highlighting the different features of the app itself. Wouldn't you it? We actually won. All in all, you know, 10 out of 10 in terms of hackathon, the food was excellent, the socializing was a blast and probably most importantly, I made some connections that I'll have for the remainder of my career and probably my life.

My next hackathon is coming up in November at Amazon Boston with Girl Develop It. I found that hackathon through Meetup and really just wanted to challenge myself to try to apply everything that I had learned from the first hackathon years ago and see if it improved my experience with hackathons overall. I'm really looking forward to the environment and I think I

have a bit of a natural preference for women's hackathons for a few reasons. I'm going to go over in a moment but I feel like I'm more than prepared this time around.

Even though I'm an entrepreneur and I don't have a ton of time to participate in hackathons, I could honestly say, I still do them because they're fun and challenging. The moment they stop being fun and challenging is the moment I'll stop doing them. I mean, I'll probably very likely speak up to help guide others who organize hackathons to help make their hackathons fun and challenging again.

On that note, let's switch quickly and talk about women's hackathons. Hackathons, as well as most of tech, is a boy's club. Women, myself included, have a lot of issues with feeling included and welcome in a lot of different tech environments but especially at hackathons and that's really important to note. I am definitely treated differently than my male counterparts as a web developer and entrepreneur who is also a woman. When you're anxious about fitting in and feeling like you belong it's really hard to focus your attention and your emotional energy on developing a great idea.

As a result, over the past years, a lot of women's hackathons have popped up. But, you know, they're still in the minority. My first hackathon, as I mentioned, was a women's hackathon and I felt that they really did an excellent job in curating an environment where women could go and boldly share their ideas without fear of judgment. Let's face it, a lot of the issues that are important to women and non-men are not important to men.

While we do absolutely need to be in those co-ed hackathon environments, sharing our experiences, what if a woman never makes it to a hackathon because of hackathon's perception of being super competitive and aggressive and she just wants to have fun and build things in the supportive environment? Men are often already comfortable with sharing their ideas, but it's an area where women frequently need support in order to flourish. Women's hackathons provide a safe space for women to go to gain that confidence so that they can go forth and share their ideas outside the bubble, so to speak.

If you're a woman who identifies with the feelings of anxiety while considering hacking in a coed environment, you're not alone. We're not weak, we're not too sensitive and you'll find that

women's hackathons actually occasionally offer accommodations for child care too, since that's an important need that women in tech have.

Many women's hackathons, are often also open to transgender, none binary and gender none conforming people, which provides an environment where they're less likely to be judged and they can develop their skills and confidence with the help of other people. Don't worry about whether or not folks agree with the concept of women's hackathons or, you know, if they think that they're less than in some way. The organizers of these events are well aware of the myths of women's hackathons and they work hard with every past to improve them and make them more inclusive.

Also, as a final note. Women's hackathons also tend to be one day events so accommodations are easier to make for people with families. Many women just don't enjoy the idea of spending an entire weekend fueled by caffeine and would rather work around adequate self-care conditions to ensure that they produce their best work. I am personally one of them. Sleep is life. So here can you find women's hackathons though? You can look on Meetup.com where they're typically opened to the public. There are often organizations that typically run hackathons like Girl Develop It, so you can see if they have local chapters and subscribe to their newsletters.

Now that I've touched on my own hackathon experience and a little on women's hackathons, let's talk about how and why you should leave a hackathon a better developer. First, you're going to build your people skills. If you're pushing your comfort zone by participating in your first hackathon at all because you're an introverted person, know that this is one of the best ways that you can push out of your comfort zone and dip your toes in the water.

Everyone at the hackathon is there for the same purpose: to meet other people, learn from them and exchange ideas. If we wanted to program alone, we could program at home, right? So strike up a conversation with someone at lunch. Ask if you can sit at a table with folks you don't know. You'll probably make new friends and probably even some new supporters when it comes time to present. At the We Code Hackathon, some of the folks that I had talked to over a snack break, erupted in tears when our team came up, which was amazing and I mean, it gives me

goosebumps to even say that out loud. That's a sign of a really, really great supportive hackathon environment.

Second, you're going to learn a lot about the value of time. Hackathons encapsulate this experience that usually takes months or years and fits it into a very short period of time. Whether you work for a company, freelance, or even if you're an entrepreneur, learning the value of time and how it affects your bottom line is incredibly important. If you can save a company time, and we all know time is money, then you'll become a very valuable asset to them. If you can save yourself time in setting up a project, you make more money for less work.

As a recovering perfectionist and chronic procrastinator myself, I can tell you that anxiety and perfectionism wreck any sense of time that you may have. Being afraid of doing something perfectly causes you to spend a lot more time on it than you should, which can eventually lead to you not wanting to do it at all. So hackathons push you to manage your time a lot more effectively and you'll benefit from it in the long run.

Third, you'll take a project from idea to prototype. This part of the entrepreneurial equation in particular can be the most difficult for some people. I've always had tons of ideas as an entrepreneurial person. But actually figuring out how to take them to a reality is a skill that you need to develop all in its own. There's a reason that hackathons focus on this particular part of the process; it's not something your typical developer faces.

I'm an entrepreneur, so I'm used to bringing my own ideas to life. But for a lot of folks who aren't necessarily creatives or worked building other's products, that's not an experience that they get in their day to day and for this reason the experience that you get at hackathons is invaluable. Everyone should have the mindset and the skills required to take an idea and make it fly. But unfortunately, too many developers they only work on projects that they're given through their e-learning platforms if they are learning or their companies and their clients if they're working.

Personally, I found that I was the person in the room with the most strength in the area of making ideas fly and I think that played a big part in why my teammates felt so comfortable taking on the idea. So be that person; put your ideas out there and add the "idea to prototype skill" to your resume. It is so, so valuable.

[BREAK]

[0:20:08.6] NA: So I am going to help you all out today and give you some solid advice on how you can get the most out of your hackathons as well as how you can prepare for it. When you go to a hackathons, things can get pretty overwhelming pretty fast if you don't go in with goals to keep you focused. So I totally get it. I had no goals my first time through.

As I mentioned before, I created this handy checklist called *Your Trusty Hackathon Checklist* for you to use in preparing for your hackathon and you can even bring it to remind you of what to focus on while you're planning your project during the hackathon. Share it with your teammates even. So stay tuned to the end of the episode to download that checklist, and then you can go register for your first hackathon and come back and tell me all about it, deal?

So first off, here's how you're going to get the most out of your first hackathon. To get the most out of your hackathon you need to know why you're there. Why did you register for this? What do you want to learn about programming or web development? What do you want to learn about yourself? What do you want to improve about yourself as a web developer? Who do you want to meet? Is there any way that you can leverage the skills that you need to learn better by having others help you with implementing them into your project?

Like if you are not so great with Flexbox or CSS Grid or working with libraries or frameworks or API's, is there any way that you can leverage that by working it into the project? How can you position yourself to observe others working while getting work done yourself? There are so many questions, so you need to know why you're there and that's going to lend itself greatly to developing your approach to getting the most out.

Second, you're going to need to network. If you're looking for a new job, networking is still far and wide the best way to meet the people who can line you up for your new job. Bring business cards. If you don't have them, get them made. Even if they say "newbie coder" on it, you know be prepared to tell people who you are and what you're doing and give them your card. Add people on LinkedIn and Twitter and just generally tell them what you are up to.

I and many others have called it the elevator pitch and it is basically a one sentence description of what you are doing. Just long enough for like an elevator ride during a casual chat. You can create a hackathon Slack chat also for sharing links, and memes of course, and just connect to people in general. Ask questions, hear response, see if you can meet up with them in person if you don't know who they are in person, you're in the same room but network.

Third, you want to leverage your fellow hackathon attendees as an impromptu user group. At my first hackathon, we had several people come into the rooms to ask questions about problems that we had faced. So be helpful and honest. Issues that people can closely identify with actually maybe more likely to be favorable if you are interested in winning, if that is a thing for you. I am going to be carrying that idea to use people to pull them into the next hackathon because it seems like an incredibly important step that so many people overlook both in a hackathon environment bringing an idea to life and in real life, unfortunately, when there is a lot more money and everything else on the line.

So next step, you want to forget about imposter syndrome. You have no time literary. Everyone there has their own strengths and weaknesses. Everyone there has their own reason for being at a hackathon too. So talking to people is going to be your best way of overcoming imposter syndrome and plus, I actually have my own feelings about the entire concept of imposter syndrome and how especially for women it's very existence can be more harmful to us than we think.

So leave your imposter syndrome at the door. You belong to whatever hackathon you register for. If you really need the reassurance of this fact, then go and talk with some of the event organizers. They should introduce themselves during the opening ceremonies and let them know that you're not feeling comfortable and let them tell you that you belong, that your experience is enough, and that you are enough. If they are good organizers and it's a very inclusive event, you should feel a lot better after communicating your fears to them.

Next step, remember your strengths. Before you go to a hackathon, have a good idea of what your personal strengths are and I don't mean as a web developer but as a person.

If you have business experience and good business sense, those skills are going to be desperately needed when it comes down to identifying your target audience, packaging your idea into a sellable product and marketing it effectively to the judges, to the audience. As a team member with business savvy, you don't even have to take your hackathon idea the whole nine yards and actually consider things like whether or not it is really a solvent idea. That isn't even a part of this and it's refreshing, honestly.

I'm definitely going to be in that seat as an entrepreneur, but I have the additional bonus skills in front end web development. So you see my business, you know my skills, my business skills are probably one of my biggest skills there and my coding skills since they are good and they are solid and I probably know things that other people on the team won't, those I tend to see as my secondary skills until they are needed. I also have graphic design skills and developing an eye catching logo or assets for your design are other much needed skills.

So creatives will have a lot of fun at hackathons especially now that you know it's not all about programming around the clock. You could be developing a dazzling user experience that tells people you understand your users but may need a little bit more time than hackathon affords in fleshing out on the backend. You need to know how to kick start your project. Whether you're developing a web or a mobile app, you need to know how to get started ASAP.

With several people on your team, this usually shouldn't be a problem but you still need to know what your options are in order to be able to discuss and agree and decide on which ones to use. Remember, the less time you spend obsessing over the details the better and that's coming from a recovering perfectionist. So use frameworks and libraries, code that other people have created. Don't recreate the wheel and consider what libraries or frameworks you might need to bring in your particular ideas that you're bringing to the hackathon. Think in advance what those might need to be able to come to life and they may also come in handy for someone else's ideas if you happen to know a library that would work for their project.

As the next idea, walk around and breathe fresh air often. Stale nerd air can make you a little bit loopy. You don't have to stay in the building, get some fresh air, stretch your legs. You'll spend about parts of time, ideally, on design and development of your product, which actually use very

different part of our brains. Exercise and oxygen can help you stimulate your creative brain and make your development path clearer when it's time to switch gears.

So what can you do in particular before the day of the hackathon to prepare? First, make a list of what bothers you about the world, what really grinds your gears? Because of it bothers you the chances are very good that it bothers someone else too. Bring those ideas to the hackathon table while you are deliberating your ideas as a group and really focus in on the pain points. How does the problem make you and other people feel? What does it keep you and others from accomplishing? I am going to be probably bringing my iPad to the hackathon to whip up a quick Google form and see if people can answer some basic questions to help us develop the app around their frustrations and needs.

Second, get your resources together beforehand. Do you tend to need help with a particular language or framework? This is actually a slightly different approach than you may thought I was going to take when I said to get your resources together. It's not just for yourself. Do make sure that you have a bookmark list of your frequented resources so you don't have to waste time looking up answers.

In that way, if you don't have an answer during the hackathons, you can show someone else that you tried to find the answer but couldn't. They will be able to be more helpful since they know what you've already looked at. Apart from leveraging resources to ask questions and learn though, you will be grateful that you have your resources together when you need to look up or remember something in particular.

Third, talk to others who have participated in hackathons and I made this one easy for you because you've already at least connected with one person who has been to a hackathon; me! So feel free to leave me comments and questions on the episode page at Lavieencode.net/21 and don't be shy. If you have any hesitations in regards to registering for hackathon then you really just have questions that need to be answered.

So you feel less insecure about actually doing it. It's all a matter of perspective. Sometimes being uncertain about an outcome keeps us from ever trying something new and fun and I know

this because I still face it every time I write or I press record on the podcast. What if you guys don't like it? What if it's not relevant? I don't know what the outcome is going to be.

But I do know one thing for sure, the only person that would be hurt by my not trying, is me. So ask around and let your developer friends help you gain the confidence to show up on hackathon day armed with your ideas and your resources.

[BREAK]

[0:31:28.6] NA: Look at that, we made it. Hopefully you've gained a different perspective than you had coming in on hackathons now and if you haven't participated in one, you are ready to sign up for your first. If you don't feel like you are quite there yet, that's okay too. Hackathons are not for everyone but for people in tech who want to expand their skill sets and put themselves into a unique situation they can be a really exciting way to both grow as a person and a professional.

If you're ready to download your trusty hackathon checklist, you can go over to the episode page at lavieencode.net/21 and download it there. Have you ever considered participating in a hackathon but felt like maybe it wouldn't be enjoyable? I want to hear your feedback and your stories on social media. So, you can leave comments on the podcast episode page at lavieencode.net/21. You can tweet me @lavie_encode or you can post on the La Vie en Code Facebook page at lavieencode.net/Facebook.

One quick announcement regarding the presale for 30 Days to Web Development, my online prep course for self-taught web developers, the pre-sale card will be opening very soon and I am so excited to get my pre-sale beta testers in and take the course. 30 Days to Web Development will be offered for 50% off, the lowest price this course has ever going to be offered for just for this VIP pre-sale students, and there's only a limited amount of space. So if you're interested in receiving pre-sale notifications, you can take advantage of this amazing one time offer and you can sign up for those on the website at 30daystowebdevelopment.com.

After the course launches, there will not be anymore 50% off discounts. So this is your best opportunity to enroll and you're going to get so much from this course, folks. You all see how much work I put into these podcast episodes, right?

How much meat I pack into every episode? Yeah, imagine that in an online course. You are going to leave 30 days to web development after just 30 days as the name implies. A transformed newbie web developer. You'll be learning faster, more confidently and with far better retention, you'll know what resources work for you and which don't, you'll know what your ideal career in web development looks like. You'll know the skills you need to get there, you'll have an internal compass that your self-taught peers who haven't taken this course absolutely will not have.

You'll feel more confident expressing your skills and benefit to potential employers and clients. It's going to culminate at the end of the road in a shorter transition time between careers, less stress, more happiness and enjoyment and best of all, more money, because you're going to know your worth. So don't be left out and miss your one time chance to get 30 Days to Web Development for half off the lowest price it's ever going to be. Again, you can sign up for the presale notifications and get your updates when the presale opens at 30daystowebdevelopment.com.

Next week, join me for an episode filled with actionable advice to help you get and stay on top of your coding education. I am going to help you develop the tools to pick yourself up and get back to coding when your motivation hits zero. I know this is something we all struggle with, myself included. I mean, this episode isn't going to even be particularly unique to self-education but I noticed that a lot of self-taught web developer struggle with this area in particular. So we'll pick apart the concept of motivation and look at what it means for you as a self-taught web developer.

Let's meet again next week on Tuesday at let's say nine AM eastern, sound good? Okay, deal, see you then. Until next time my friends. Peace, love and code.

[END]