projectorize instructions-Initiation steps

I'm going to talk about either the Module Manual or Projectorize or both, but I'm going to go ahead and do, I guess I'll, I guess I'll pitch Projectorize first. Do you have the thing running? Yeah. Okay.

So this, what is Projectorize? Projectorize is actually some code that, that, well, it's a, it's a project management style. Okay. And it comes with a lot of benefits that the reason we're recording right now is because I want us to list those benefits out and actually make a document on why, who, what, when, where, how, why on Projectorize.

Okay. So that people understand why it's so critically important to write the project plan in this particular format. There's a lot of benefits for that and I want to list those off so that people understand that this is like non-negotiable, like this is one of the swords that I'm willing to die on.

It helps us because we can automate a lot of tasks related to project management if we have it in this specific format. Right. So if you look here, I have the project, okay, Projectorize is a project management framework or a system that allows us to communicate easier.

So how does it work? So you have projects and inside your projects you have phases to complete the project. As you can see here, there is a project and I was able to just type Z-O, well Z, whoopsie, I'm able to close the project like this, meaning to fold the text. What does fold the text mean? It means to be able to hide.

Collapse it. So here, using VS Code, I can just simply type Z-O. And this fact of this user interface is not critically important for everyone, but it is critically important for you and I, or especially me.

Because my goal, my biggest pain point is that I know a lot of subject matter stuff over entrepreneurship, finance, investing, software development, system administration. I happen to know a lot of jargon. And when I bring people on the team, I'm spitting so many concepts at them.

They don't have the ability to take notes. They don't have the ability to tie everything together. So I need to communicate better.

And the way that I can communicate better is rather than talking to people as much as I do, what I can do instead is put everything that I know in this format, not only explain the concepts, but explain why you need to do things a certain way. And as well as put all the details. So I put who, what, when, where, how, why on each element of every task, every stage, every phase, and the project itself.

So that I can send this document to someone and they can see the entirety of the detail. So now let's get into features. So right now, and I really hate the word phase.

I don't hate it. I just don't love it. But one thing is I would rather there not be two Ps.

I would rather it be project and something else. But this is a minor optimization. But basically the project is written like this.

So you have the project and inside a project you have different phases in order to complete the project, as you can see here. Then you have the, oh, you can expand a phase and you can see a one sentence explanation or a high level view of all the stages needed to complete the phase. So it's a to-do list inside a to-do list.

And then you can go into any of the stages, Z-O, open it up, and now you can see all the tasks, one sentence on what those tasks are in order to complete the stage. And then you can open up each task. Boom.

And it tells you who, what, when, where, how, why, which is six different sentences around how to complete the task. So in the header of each of the project, phase, stage, and task, then what I do, and we'll just call those tags just for making it easier, because every single sentence in this document is tagged with either project, phase, stage, or task, or it's tagged with who, what, when, where, how, why, or notes. All right.

So what are the benefits of doing it this way? Well, first of all, we can transform this. So if this document, which is currently 5,000 lines long, is overwhelming, what we're able to do is automatically go through this file and remove all of the how, why, where, and when, and who. We could just remove it because it's tagged.

We can just simply redact it from the document and send people the top-level view, which would just be the project, the phases, the stages, the tasks. And what? And, huh? The project, the phases, the stages, the tasks, and what? Are you being, take out all the what, how? No, you can leave the what. And you could leave the what and the why and take out the how, the where, the when, the who.

Okay. You could take out anything you want to take out. So it is filterable.

Yes, because you can literally automate it and rip it apart. And the benefit of that is, so for example, I know a lot of, which is weird, I know some jargon and I don't know other jargon, but the best policy is to go and get the jargon. And one of the rules about Projectorize is you intentionally put the jargon as the stage, phase, task, or project.

You intentionally put the jargon there. So here you can see apply for an EIN. What is an EIN? So you, when you open it, then you're putting the defining thing of what an EIN is.

That way you're teaching your team the jargon. Okay. Because your jargon sits in the header

and the what is the definition.

So you can see the EIN is an employment identification number that is issued by the IRS. That is what it is. Uh-huh.

And then how are you going to apply for an EIN? Well, you're going to learn about the EINs and apply for an EIN using a U.S. government website. And why are you going to do it? Well, it's used for taxes, but why we need it specifically is we need to sign up for the EIN so that we can get access to the payment gateway so that we can move money. Right.

So you have to do this. And so we need it for the taxes, but we need it for the payment gateways and our money system. Uh-huh.

And it tells you why you need it. Where can you find out more information? Here we can put links to our knowledge base or to the internet. Where wouldn't be in this case would not wouldn't where be the website to apply for the EIN online in this case? Well, no, because in unfortunately, well, I did this a little bit wrong, but notice how the stages apply for an EIN.

And down here, the task is apply for EIN. So it's in there. There's your freaking link right to where you need to go.

But it's important because apply for the EIN. Right. The IRS has a website for applying for the employment identification number.

So I'm typing it out. And then I'm saying click on apply for an EIN online button and complete the application where at this specific link. Why getting EIN is the first step in creating a business entity and to give us access to the payment gateway that we need.

If we don't have the payment gateway, we can't move money. So the fact that this and this are named the same is a little bit lazy, I guess you could say. But you could do the first one as prepare.

No, no, no. So the stage is acquire. You could say, well, no, that would be acquire.

OK. And I am right. So you're getting an EIN number.

So you've got to apply here. That's your task. And as you're applying, you're going to select, well, start applying for an EIN.

Right. And as you're starting there, then you're going to as part of it, we're going to tell you to use an S-Corp. And then we're going to explain why we're picking an S-Corp.

Why are we picking an S-Corp? Why? Are you asking right now? Yes, I'm asking right now. Because it has the benefits of a C-Corp and the benefits of an LLC. For example, it has flow through taxes.

So here you can see S-Corporations enjoy flow through taxes and yearly tax filing. So if you do a C-Corp, last I checked, you have to do quarterly filing of taxes. OK.

And secondly, if you take in money and you give it to someone else in a C-Corp, you can be taxed for the money that you're taking in, as well as that person can get taxed when they receive the money. So there's double taxation with an S-Corp. You can just simply take the money and pass it over and say, well, we're not liable and they get taxed and you don't.

And then you don't have to worry about taxes so much. So all of this stuff I've actually thought through, done like hours of research and I'm like, OK, we're going to standardize on the S-Corp. Of course, I have to do a legal disclaimer and say, well, you have to talk to your own people, blah, blah, blah, blah, blah.

But the fundamental thing is we're going to be building this entire standard operating procedure around reducing costs, which includes filing quarterly paperwork. Right. So that's why you do S-Corp.

Then the other thing is authenticate and then enter the state. And then this is one of the most important things way down here. Shoot.

I recommend the Delaware, but we could change it later. Somebody told me Wyoming. It varies, but we don't need to worry about that.

Actually, I'm way off topic here. We should be talking about projectorized. So this even talks about how you get your EIN letter.

And later somewhere down in here, there is a special ID number classification that tells you what the classification. Oh, that's when you're incorporating. It tells you what your tax classification number is.

I forgot what that's called, but it's in here and I forgot, but it's in here. So, again, this serves as a knowledge base and it explains that you're going to based on this module manual, the activities that you're doing for business, what that classification probably is. Right.

So now you can see the stages that are easily in there. And because it's written this way in the phases and the stages, it's easy for somebody to get a top level view and literally read it from top to bottom in 10 minutes. And just boom, like that.

If they just read the stage view and ignored all the tasks in each stage. Right. Another benefit for projectorized is that I've been trying to work with non-technical people.

And they're like, well, just I love your technical idea, but I'm not a coder and I love your vision. But what can I do as a non-programmer? Well, when we give them the module manual, we'll be able to say, can you do this? So one person has like a team of 10 people. And we could say, OK, give this to your team.

First, we ask him, OK, do you understand it? Yes. Great. OK, give this to your team and then they can ask questions and we update the manual itself.

That's how projectorized works. Another thing is I get sometimes frustrated when people forget or they say they don't know. So the easiest way to deal with that is we can teach these people to take a block of text.

And because we've given the context, anybody can now copy what I've written, paste it into perplexity and get a five paragraph fucking explanation for what it is, why it is and even get opposing views and it can teach them. Another reason for projectorized is actually this is the outline and a lot of the content for a book. So I've been taking a sales playbook.

If you look for the word playbook. Then you do it again. It's right here.

So here you can download the sales hiring playbook. What I've been doing is a PDF. What I've been doing is reading the book, which is written in paragraph form and transforming the book into actionable stages, phases and tasks.

Right. And that means the book would be useful now. Then we can take everything we know about sales, which is specifically recruiting, interviewing, hiring, managing, onboarding, managing, compensating a remote only commission only sales team.

So that's a very specific freaking thing. And then what we'll do is once all that's written out, we will go to the sales subreddit and we will go and find sales managers, expert sales managers and have them review the projectorize of this with all the detail and say, where's the lie? We can go to individual salespeople and do formal customer discovery and say, which I've done to an extent. So I signed up for all these sales subreddits and I did a deep dive into sales culture.

And then I started figuring out what they hate, what they like. And I started to plan out the things that they want based on their words, what they are looking for as a salesperson. And some hate commission only and some like commission only.

So there's a big divide within sales culture. You do not have to pay a salary in order to reserve high quality, high performance sales people. And some of the sales people will even admonish the ones that require a salary and say, you're just not willing to hire yourself, which is the argument I would use.

So I I've seen both. And it's the same with software developers. Some people need to be paid.

But the entirety of the Internet, including this V.S. code is open source. This manual is open source. The micromanager is open source.

So you you're literally sitting next to somebody who would work for years for free and give it away. Like that is the reality of our actual not just anecdotal, but the entire everything you see on the screen was built on open source. Everything here, the operating system, the V.S. code,

even YouTube started out with and probably still and still operates on some open source software.

And the browser behind it is open source. There is nothing. And you've got to understand that there are tens of thousands, thousands of applications underneath every single one of these things.

So and all of that, most of that, the lion's share of it developed for free. So we want to give people an opportunity. And as a business owner, you want to give people just pretty much percentage of the revenue or percentage of the profit straight up rather than trying to, you know, run yourself in the red and stuff like that.

And what I mean by that is like a squeeze. In other words, pay people less than what they're worth. I would rather pay people what they're worth.

And if they're worth nothing, then it shows and they don't get paid because I don't get paid if the company doesn't go well either. So I'm going to extend that same risk down to the people who are promising that they can be competent and actually execute tasks. Right.

Oh, I don't know why there was a missing. Start applying for EIN. I don't know what happened to that text over there.

That's why I did a reverse on it. All right. So now let's talk about more benefits of Projectorize.

Because everything is formatted this way, we can transform it, meaning that I can easily do a split file. OK, so there's a command over here. I forgot.

I don't need to get in the technicals of it, but there is a system command line prompt. I forgot. It might be split.

Right. So split will allow me to take this file and split it out by task stages, phases. Everything would be in a separate file and numerically incremented so that you can see the entire plan, but in multiple files.

And what that does is it means that we can use this plan as a series of prompts in micromanager. So by taking this phase and passing it into the large language model, let's say you wrote three sentences for the what or you your sentences too long. So it's wrapping all over the place.

Right. We want concise communication in a certain tone and style with professionalism and accuracy. And what this does is it allows me to actually call the model on every single who, what, when, where, how, why on every project, phase, stage and task.

And have the large language model automatically edit it and enforce certain rules. For example, one of the rules you can just see that jumps out at you right here. And I know I'm a little bit of a perfectionist, but the thing is, like you can go here and notice that all of the phases are

specifically starting with verbs and they must be three words.

Review the overview, start the systems, raise the revenue, make the module, manage the money, learn the leadership, hire the help. And you can even see that there's alliterations. R-O-S-S-R-R-M-M-M-L-L-I-H-H-M-M-S-S-M-N like that.

So you can see that I'm putting a lot of extra thought into writing this and my perfectionism can slow me down. You mean OCD? No, I think that's just straight up. I think it's OCD.

Whatever, it doesn't matter what it's called. The fundamental point is that the clarity and simplicity and the conciseness of this and the pattern. I want people to learn the plan and know that the plan is the highest authority, that it is the boss.

I don't necessarily like bossing people around and I don't like them bossing me around. What I would rather is that we have a collaborative environment where we all create the plan, we all vet the plan by taking it to subject matter experts, we all actually execute the plan. And as we learn insights on the right way to do, we tweak the plan.

It's a living document. As interfaces change, as laws change, we tweak the standard operating procedure around those changes or to make things more clear, right? But as you can see in this plan, a lot of it is not filled in. So if you look here, for example, well, actually you don't have to go there.

You can go here and say Z, O, and now everything pops open. Now you can see the where, the when, and the who were not filled in for this task. With regard to when, I'm actually somewhat anti-setting deadlines.

Because I believe write it out as a waterfall plan and then execute it top to bottom and measure your velocity because all of your tasks take a certain amount of time limit. So if your time limit on each of the tasks is three hours to one day, then you should know your rough estimate on how long it should do your plan. But if it's in three hour, if each task is three hours, then this 500 tasks, no, it's 5,000 tasks, 500 tasks can be done in under 500 days.

It's going to look more like this whole thing could actually be done in like three to four months. Like if you're sitting there eight hours a day, you're going to blow through this real quick, like literally 30 days. It's fast because it looks like a lot, but think about if you're following along with 270 directions on how to set up and configure a CRM, that's two to three days.

Like you can do it in one sitting. I shit you not. I've done it in one sitting in 20 minutes because I knew I followed my own directions.

Then I gave it to other people. It took them two to three days. So from scratch, from cold.

Right. And that type of without asking me questions. So it's detailed enough that all the context is there.

Right. And it's reusable. So one of the other benefits of Projectorize is that it is when people flake out, you can easily replace them with a different person.

Right. So I'll have people come up and be like, yeah, Drew, by the time you get all the work done, I would love to be a part of your project and just cash in as you start to make money. But this allows people the opportunity to actually partner with you.

And it defines specific scope. Yes. Another benefit is and this happened with Jay specifically, he was very concerned about our agreement.

He was like, well, you know, let's write a contract on what my equity is. OK, but he hasn't done any work yet. So one of the cool things about him was that by him challenging me to do that, I realized that Projectorize is actually part of our partnership agreement.

We write out all the responsibilities and the tasks that need to be done and stages and then we delegate them. And that person will say that they have a clear understanding of it and they don't get to earn equity or a percentage of revenue if they do no work. If they can't complete a stage, honest to God, fuck them.

I don't care at this point. I'm just not allowing people to to promise and just under deliver because I'm held to a higher standard. We all are based on the needs of the business, not the needs of the person.

Right. And we'll get into we can get into the whole Agile manifesto. I think everything on that Agile manifesto is backwards.

It puts people over processes and I say, no, put the processes over the people. So I'm like flipping everything. Now, what I would like to do is focus my energy on strategy, research, development, knowing the inter dependencies on all of these different concepts and write out Projectorize and actually separate myself from working directly with people for a while.

So that I'm pushing instead of teaching Graham and training him up and spending all of that time talking to the different team members and trying to pour that value into them. I want to pour it into the Projectorize plans and then the people who want to work with people more or first of all, make me a lot more efficient because instead of me repeating all of this type of stuff, I can just email it. Right.

Secondly, and it means that people won't be able to let me down as much because they'll be able to like they either did the stage or they didn't do the stage. Here's the assignment. I don't want them calling back 10 days later talking about, yeah, I haven't done anything yet.

That's been happening. Right. So now I can just be like, OK, this is the agreed on deadline.

And also because we're able to use the large language model, we can even get it to do some of the estimates because it can like figure out, well, this based on all six of these sentences in the

context, it can kind of look it up on the Internet or guesstimate the amount of time it should take. And then I'll make micromanager actually spit out a date. And when those dates are wrong, we can destroy all the wins, run it through micromanager again and get new estimates on 500 tasks, which is awesome.

Right. Another thing that Projectorize helps us do is it allows us to take this format because it's formatted this way, we can easily convert it into JSON and inject it into other project management systems like Odoo, SweetCRM has its own project management system that nobody knows about, Bittrex, Salesforce and spreadsheets, which to me is I mean, the spreadsheets are a lot of people use spreadsheets and that and there's a good reason for it. But another thing it does is it decouples us from Bittrex, from Google Sheets.

So I'm decoupling myself from Google because I've been locked out of a Google account before and I'm currently locked out of a Google account right now that refuses to give me access because I no longer have a cell phone number that I gave up seven years, eight years ago. So the next thing that it allows you to do is so yeah, so we can use automation. Plus, these work out to be prompts.

Okay, in the future, we're going to be able to have a stage or a task and say implement this tool. Here's where to find the tool. So you can go over here under micromanager.

Under micromanager, for example, and what we're going to do is say implement priority one software, right? There's reasons for that. So zero open, I mean, Zed open, that's to open this. And now you can say, I want to actually implement different software.

And so we're going to write this up. And the who that's supposed to do it will be micromanager itself. And we will pass in implement this software.

Why should we do it? Where do you find it? When should we do it?

This file is longer than 30 minutes.