**AI Acceleration (Towne Park) Backlog Grooming-20250630\_203453-Meeting Recording**

June 30, 2025, 7:29PM

1h 5m 25s

 **Jonathan Aulson** 0:23  
I guess.

 **Andrew Scheuer** 0:24  
Hey everyone. What's up?

 **Javier Casas** 0:28  
Hello.

 **Andrew Scheuer** 0:51  
8.  
I mean, what you do on your vacation?

 **Javier Casas** 0:57  
I love the sunbathing.  
And I don't know, traveling, eating, enjoying my parents visit.

 **Andrew Scheuer** 1:09  
Oh, they came to visit you.

 **Javier Casas** 1:12  
Yeah, they're staying here until 8 July 8th.

 **Andrew Scheuer** 1:19  
I got a is your your your wife's family is there too, right?

 **Javier Casas** 1:26  
And no, it was my mother-in-law. She came here before my parents do. So yeah, full house. Yeah, full house for June and.

 **Andrew Scheuer** 1:29  
Mm.  
So you got, so you got payback.

 **Javier Casas** 1:42  
Of July.

 **Andrew Scheuer** 1:44  
Well.

 **Javier Casas** 1:49  
We missed them, so it's it's OK.

 **Andrew Scheuer** 1:54  
Yeah, no, I mean that. I don't know, maybe just my family, my the first few days are great and then afterwards it's.  
You remember why you moved out?

 **Javier Casas** 2:06  
Yeah.  
Kind of.  
But you know that there's a time limit. Sorry.

 **Jonathan Aulson** 2:12  
So, guys, oh.  
No, you're fine. It's it's fun hearing about your vacation, Avi. So guys we have we have a few topics I think today for this call and so I kind of wanted to well one, I'm I'm just ask John if he's going to join because it seems like it's a good meeting today.  
So we might give him a minute, but I think we have Andrew, you had something I think you wanted to show. I had kind of put Graham on the spot asking him to show something and then I've got a couple topics. Does anyone else have any any stuff they want to go through on this one?

 **Javier Casas** 2:52  
Yes, Sir.

 **Jonathan Aulson** 2:56  
OK, fair enough. So let's try and keep our shares to like somewhere around the 10 minute mark. Oh, John is going to come, so cool. So maybe why don't we start Graham, if you're if you're up for it, if we could take a look at.  
The trouble you ran into and maybe understand more or or off comments that I think think that'd be a good one to start with.

 **Graham Olson** 3:18  
Yeah, yeah, I'm not. I'm not entirely sure on how to share this. I can let me, I can try to like recreate kind of what I did and maybe the errors that I got.  
I.

 **Jonathan Aulson** 3:34  
Yeah.

 **Graham Olson** 3:36  
But essentially, uh.  
All right, let me share.  
Can you see my screen here?

 **Jonathan Aulson** 3:47  
Oh, there it is. Yeah.

 **Andrew Scheuer** 3:49  
Yes.

 **Graham Olson** 3:50  
OK.  
So essentially what I start a new task.  
What I did was try to give it some context for so like um.  
Let's see create.  
Uh, unit tests.  
Or.  
The labor hour calculator.

 **Jonathan Aulson** 4:23  
You know, I'm sorry to stop you, Graham. I wonder, is it does it, does it happen to be in your recent tasks list like the one you're looking, the one you're reproducing?

 **Graham Olson** 4:26  
Yeah.  
Dude.  
It might be.  
June 30th. That's all today.  
I don't know 'cause it's not gonna show me like like what what I was specifically seeing was it was essentially as if the IT wasn't really getting the context. So what I provided was.  
The class that I wanted to test in this case it was the per labor hour calculator, so I gave it that file to specifically look at and then I I know that Andrew has some tests in here for some other calculators.  
Um.  
For like revenue share, I believe was what I was kind of basing it off of. So I was giving it that context as well. Um.  
Yeah, like this revenue share calculator, um, since they're kind of, you know, they're similar in.  
Logic kind of and structure. They're both using that Internal Revenue interface. So, so I was giving it this as like a example of maybe how to test, you know and it.  
It seemed like it just wasn't, no matter how many times I iterated on it and provided the errors that.  
That I got it. It just was not resolving them, which just led me to the.

 **Jonathan Aulson** 6:25  
Mhm.

 **Graham Olson** 6:28  
You know, do it just by hand and create the test. So I guess my question is, has anybody ran into issues with with testing with client I was using?  
4.1 uh.  
Yeah, so he's using this language model. It seems like tests are seem like the best use case for AI like we've talked about and in the past I feel like that's the case, but.  
Not I don't know. I don't know what the issue was.

 **Jonathan Aulson** 7:07  
E.

 **Andrew Scheuer** 7:08  
Did you start? Did you start a new task in order to start this or were you already working on on this this feature in the same conversation?

 **Graham Olson** 7:23  
I think I I think I started a new task. Um, yeah.

 **Jonathan Aulson** 7:26  
It.

 **Andrew Scheuer** 7:29  
Yeah, um, so.

 **Graham Olson** 7:31  
So could that be the issues like it doesn't have the context maybe of the?

 **Andrew Scheuer** 7:35  
Yeah, yeah. I found that giving it the the user story, the acceptance criteria and having it look at the if it if there is a commit, look at that commit that can give it like a better idea of what to test.

 **Graham Olson** 7:49  
Good evening.

 **Andrew Scheuer** 7:53  
Um.  
So, so I've had, I haven't had this happen to me where it hasn't been. It's kind of got stuck in the loop or stuck in a rut, but I I know when I whenever I'm doing like an isolated so feature that's already been completed.

 **Graham Olson** 8:03  
Mhm.

 **Andrew Scheuer** 8:11  
And I want either my unit test or I want to do something else with it. I kind of have to give it that context in order for it to know you know where where things are and also kind of explaining it just the just the more context you can give it.

 **Graham Olson** 8:12  
Mhm.

 **Andrew Scheuer** 8:26  
The better.

 **Graham Olson** 8:28  
Right. OK, that that makes sense. Yeah, I think context is is pretty key to using AI tools and I think that's maybe where the issue was, was.

 **Andrew Scheuer** 8:39  
Yeah.

 **Jonathan Aulson** 8:41  
Mm.

 **Graham Olson** 8:42  
Was with that. Um.  
So in the future, that's good to know. So you're saying using the same task as.  
Like when the code is implemented, just using using that same task and getting the tests going on.

 **Andrew Scheuer** 9:01  
Yeah, or even like if if it's a PR that you did any any place where you can you can show it where the code specifically was implemented or just pointed it to the right file with the user story or the task that that might help.  
And and getting a better outcome.

 **Graham Olson** 9:17  
OK.

 **Andrew Scheuer** 9:21  
Yeah, otherwise like it wants to please you, especially GPT 41. It it wants to like give you an answer. So sometimes it'll it'll lead you down the wrong path and also like breaking it up into steps too, like maybe looking at the feature and then pulling out one thing you want to unit test.

 **Graham Olson** 9:26  
Awesome.

 **Andrew Scheuer** 9:40  
It might help you at the beginning to see if it's going down the wrong path, if it just doesn't understand. And sometimes I'll ask it like I'll just say, you know, especially if I'm starting a new task where there's something already implemented, I'll be like, what do you understand?  
About this feature and then you can correct things.

 **Graham Olson** 10:05  
OK, gotcha.

 **Andrew Scheuer** 10:07  
Yeah.

 **Jonathan Aulson** 10:10  
That was, I think that's useful. Thank you Graham for bringing that one.

 **Graham Olson** 10:15  
Yeah.

 **Jonathan Aulson** 10:17  
Andrew, I think, I think you had something for today, right?

 **Andrew Scheuer** 10:20  
Yeah, um, so I've been looking my off time.  
At a lot of different tools and and things that that are offered because it seems like every few months there's you know innovations done to one that are better and that's why like at the beginning I was like we should really be tool agnostic and try to just.  
Use the best tools that we have available and using that markdown that we have that that makes us that we're able to do that. I've been looking at cloud code which is another it's a.  
Another A I tool, but I think they really kind of focus on just developing and the coding experience more than more than anything else. There's a couple cool features just for the cloud itself that.  
I think you might like John. So today for example I use Cloud. I use Cloud code on a on a feature and one thing that what I was able to do that I was not able to do with client is be able to.  
Incorporate dataverse into the flow and understanding of how to complete a a task because like we have in our models, we have the context of like the column names and the data type.  
But like sometimes we'll have like the data type will be a string and but it's a nested Jason and so like in order to come up with a better solution we kind of that knowing that data had at times useful.  
So I made this with Claude. There's a lot of stuff that like I'm not gonna go over, but so Claude code, it lives in the terminal.  
So it you don't have to actually like there's no extension or anything. And one of the cool things about it being in the terminal is that you can use anything that's installed in the terminal. So like any type of Azure CLI.  
You can you can use in this. In order to use it in Windows you have to use it in WSL, which is the Windows subsystem for Linux. If you have a Mac then you can just install it, or if you have Linux you can just install it.  
so like oops.  
You wanna start it up? I have a I have a a cloud subscription which is 20 bucks which is like the the base subscription and they have other ones too.  
But you can just pretty much ask it, you know.  
It's the same thing as client where you can you can have you write tests and do all that stuff. It's just it's instead of an extension you can use a CLI.  
One of the things that it has that's useful for like connecting the dataverse is it just uses the the PAC, the Power Automate CLI because I already have it installed in my.  
In the terminal, so there's no like MCPS you have to install. If you want to make a PR, you can use just the Azure CLI with Oauth so you don't have. There's no tokens or anything that you have to share.  
For the feature today I was able to you know query dataverse and implement and get that into the context and just with this with this query.  
With a fetch and it was able to to implement a solution or modify a implementation that it was already starting based on that.  
Like I said, the authentication for that specifically is like multi-face, same authentication where it will ask you to sign in and you only have read only permissions, yeah.

 **Javier Casas** 14:27  
Hey.

 **Johnn Hesseltine** 14:30  
And we can.  
Your I'm sorry, can you explain again what was actually achieved here? I I feel like I missed it.

 **Andrew Scheuer** 14:46  
Yeah, no, I think I'm going pretty fast. So I had a task where I had to do some calculations based on some data that are in 2 tables.  
And in order to see what data was already saved in those two tables, I was able to query Dataverse while planning. Well, you know, I was having a conversation with Claude trying to plan out how I'm going to complete the task and I was able to use the.  
Response from that query.  
To inform the implementation to form the planning. So for this specific one there was a nested JSON with you know some with with just like true or false flags that we needed to in order to do the logic.  
And if I wasn't able to do that, then I would have, you know, later on had to implement that, but it would have, you know, been an extra, extra step, extra iteration. So we were able to, yeah, no. So like with this I was able to, you know, cut that out and.

 **Johnn Hesseltine** 15:46  
I think what's go ahead, go ahead, go ahead.

 **Andrew Scheuer** 15:53  
It was able to see, wait, this is a nested JSON. I see there's some values here and also it was trying to like in the originally it said oh we have to make like 2 new tables because those values that were in the nested JSON that I needed for the calculation that was in the user story acceptance criteria wasn't there. So it said Oh well we have to.  
Create 2 tables, but when it saw that nested Jason, it said oh, we actually have all of that, so we don't need the two new tables or anything.

 **Johnn Hesseltine** 16:19  
Yeah, yeah. I think it's important to call out there that the utility of this particular use case or scenario was really dependent upon the fact that.  
There was structured data in that, or at least semi structured data in that column, right? In terms of the Jason, like in other words, if the values in the columns were instead just integers, you know well.  
That's like, that's not so.  
Valuable I guess to this particular use case and this is this is very specifically valuable to the case where the.  
Where the the data was in some sort of semi structured format. So then you know then to give you a recommendation or a suggestion how you might implement it. I mean otherwise it would also it would have been. It would also be helpful if instead of looking at a particular column maybe you looked at tables.  
And it was similar to the way it might reason around the structure of a Jason in certain fields. It could also reason around the columns in a in multiple tables and potentially tell you how to like join them if that's what they were you were going for. Is that?  
Is that making sense what I'm saying there? Like it's real, you know. So if it's a string or you know or some sort of integer or number, right? Maybe even a Boo like if the if that's the data type of a very specific column, then this is probably less important than really just inspecting like the structure.  
Of whether it's relational and calm tabular data or whether it's in some sort of like a Jason object. Does that does that resonate, Andrew? Or did I perhaps miss the miss something there?

 **Andrew Scheuer** 18:06  
Yeah, I mean, I mean this is just, this is the example that I used today that I ran into today. I was thinking about other, you know, use cases for this is test data, you know, we're trying to do a lot of calculations based.

 **Johnn Hesseltine** 18:10  
Yeah. Yep.

 **Andrew Scheuer** 18:23  
On data we don't have and so being able to query the tables and multiple tables in a minute and finding out having to decide you know for all these calculations we need, we don't have data for for this, this and this value in order to you know verify.

 **Johnn Hesseltine** 18:23  
Yeah.

 **Andrew Scheuer** 18:43  
That our calculations are correct. For example, that's just an another example of having a connection into dataverse. Being able to read it and use that as context could help with with our our development cycle.

 **Johnn Hesseltine** 18:56  
Yeah.  
Yeah, no, I I love it. I see. And it's just that.  
It's not so much that Claude code is providing you with that feature, but in terms of the ability to execute command line well commands, that's just a.  
Benefit of.  
Interacting with Claude code from the command line 'cause you now then now you have the ability to um.  
To um.  
You know to use any other command line, you know tool that you may have installed, right? I mean it I I guess and we're saying that client doesn't have any kind of capability to to do this type of thing like to perhaps.  
You know, in the the client experience you can't like, oh hey, let me execute a command on the terminal and you use the output as a part of my context going forward that that capability is just not there.  
Yeah.  
Yeah.

 **Andrew Scheuer** 20:12  
Use and it has like direct access to that. Whether or not there's there's a worker on or a tool that client has, I'm unaware of. I tried looking for one I can. I've had trouble trying to have it consistently go to the command line for something.

 **Johnn Hesseltine** 20:18  
Yeah.  
Yeah, OK.  
OK. And I don't mean to stop you down. You may have, you may have a ton more to go. So I don't we can keep going. I just, I was just intrigued by or I wanted to make sure I fully understood what was being said on that slide.

 **Andrew Scheuer** 20:30  
Um, yeah.  
Yeah. No, that's fine. Yeah. So again, like, yeah, it is that it's specific to our use case. I think the improvements that Claude could have.  
And again, that's as of right now, you know, and a weak client might come out with all this stuff and be like, you know, make all this mute, but like all this pointless. But another use case was that I wanted to validate all the data, for example for.  
For this calculations I was doing and it was actually because I had access to dataverse and everything, it was able to give me the the queries to go and query the data through SQL Server, which was nice that I could include that as well in the PR.

 **Johnn Hesseltine** 21:22  
Hmm.

 **Andrew Scheuer** 21:27  
For people want to verify the calculations.

 **Johnn Hesseltine** 21:28  
Hmm.  
Got it.

 **Andrew Scheuer** 21:31  
So yeah, so there's there's that. Another thing with Claude that I wanted to show you John real quick was this thing called artifacts and we use VO dot dev for mocking for creating mock ups, but you can also use Claude for that and it's not it's like it just.  
Makes it in in HTML and and I think CSS so it's it's not as tied into next JS as as vo.dev like I made this with Claude. See if I can pull up.  
Yeah.  
This is just called an artifact.  
And you can create. This is obviously a presentation, but you can create websites by using this and export them as HTML.

 **Johnn Hesseltine** 22:15  
And.

 **Andrew Scheuer** 22:28  
And you can iterate over them just like VO dot dev.  
So I'm just wanted another another tool for you to look at in the endless sea of tools.

 **Jonathan Aulson** 22:33  
It's pretty cool.  
Yeah.  
That's pretty awesome.

 **Andrew Scheuer** 22:46  
Yeah, that's it. That's it for me.  
Thank you. That's it.

 **Jonathan Aulson** 22:51  
We need to figure out an approach, I think with some because it's funny, Andrew, I have a couple of tools as well that I found that I wanted to show today and this feels like a good place to talk about it like this meeting, but it also feels like we need a platform.  
Broader, you know, and and with a group of folks that kind of can help.  
Figure a plan out to because we need to adopt A set of tools, right? I would think as as a firm, maybe it's maybe it's team by team. I don't know. Is it? Because I mean, I love that presentation. We need you should be able to give that other places, you know?

 **Andrew Scheuer** 23:36  
Well, then we have like the communities.  
There's the innovation community.

 **Johnn Hesseltine** 23:40  
Is there is an innovation community there? OK, I suspect that's where it would go.

 **Andrew Scheuer** 23:44  
Yeah.  
Yeah, I talked with Rob, I think. Yeah, I talked with Rob last week when he's in the office and he mentioned that, you know, a good place to go with this would be the innovation community.

 **Jonathan Aulson** 23:59  
Hmm.

 **Johnn Hesseltine** 24:02  
Well, yeah, I mean, there's also.  
Yeah, that's interesting. Like there's two, there's kind of two separate work streams, right? There's, well, community's not really a work stream, but it's a a place to have.  
In particular, with respect to the innovation community, it's a place to have conversations around everything under that umbrella, but there's also something a little more.  
Uh.  
I don't know strategic and and then oddly in some ways tactical at the same time, but like there's a whole concept around.  
Having an opinion and a perspective on how how we leverage AI tools in our development teams as a firm, right, which also I think you kind of alluded to. Are you starting to allude to that concept, right? It really kind of.  
Etching that in stone a little bit, I do think.  
No, I I do think the can be, you know, I don't know it's some in some ways this landscape's evolving so quickly that being prescriptive on tools sets.  
Can be, um, perhaps you know, limiting, if not just not the way to go, right? Like there may be a collection of tools that are approved, you know? Um, but I I just all that to say like and.

 **Jonathan Aulson** 25:32  
Yeah.

 **Johnn Hesseltine** 25:36  
There, there's a and something I'm having a conversation with Matt about around particularly from like a tech and cloud perspective, putting some opinions and perspectives around like hey, I mean frankly in some ways it's not so dissimilar from a later way for.  
For dev teams and AI tooling.  
Um.  
And I'm having these conversations with Matt already, but I see this team as being a part of that conversation because of the work that you guys are doing in this area.  
I don't know. I maybe I'll just stop there for a second, alright, 'cause it's kinda like, you know, more to come on that, right? What that really means, but.  
I just wanted to.  
Comment on that when I heard, you know, all of a sudden you talking about the idea of potentially memorializing some of this kind of firm wide zone.

 **Jonathan Aulson** 26:34  
Yeah, it's really for me. It's like I just there should be a plan that is kind of evaluating tools and and kind of associating tool sets with tech stacks and making that information kind of available for everyone to say like when we start a project kinda.  
Go pick from this list of cool stuff that we think is cool, you know? Yeah.

 **Johnn Hesseltine** 26:59  
Yeah.

 **Jonathan Aulson** 27:01  
Anyway, is is it? Oh, sorry.

 **Johnn Hesseltine** 27:02  
Yeah, and I think that aligns to what?  
I was just gonna say I I think that aligns, although, well, I think that aligns to what the conversations that Mack and I have had, that exact thing. We don't have it yet, right. Which is the the point. But with that, that's exactly the kinds of things that we're thinking about and I'm thinking about that I think.

 **Jonathan Aulson** 27:12  
Yeah.

 **Johnn Hesseltine** 27:23  
We should have a an explicit and coherent story around, right? An opinion around, a perspective around.

 **Jonathan Aulson** 27:30  
Yeah.  
And is it for innovation? Is is it Rob we go to or who is that?

 **Johnn Hesseltine** 27:40  
From a community perspective.

 **Jonathan Aulson** 27:43  
Yeah.

 **Johnn Hesseltine** 27:44  
Uh, that's a great question. Um.  
Um.  
Surprised if it's Rob, but I guess that's a possibility. I mean, I I can go get that answer. I don't know for sure who owns owns that community.

 **Jonathan Aulson** 27:56  
OK.  
I I can probably find it. I assume it's in the knowledge base, but I'll check it out. All right.  
Well, that was a good one, so.  
Andrew, what's your? Is your sense like, hey, we should all just, you know, we should be using this instead of Klein. Is that are you there yet?

 **Johnn Hesseltine** 28:21  
You are muted.

 **Jonathan Aulson** 28:22  
Mhm.

 **Andrew Scheuer** 28:25  
Sorry, did someone ask me that?

 **Jonathan Aulson** 28:28  
I was curious where where you're at with this Claude tool. Are you of the opinion we should be using this and not Klein or not not yet?

 **Andrew Scheuer** 28:38  
No, I mean, I I can't say that. No, I mean, it's it just is the flavor of the month for me. Like, like, I don't wanna like, I want people to use whatever they they think that they're having success with.

 **Jonathan Aulson** 28:41  
Yeah.

 **Andrew Scheuer** 28:53  
But keeping since they're all configured in kind of the same way, we can keep all of that knowledge in the repo. I'm trying all of these new things cuz I was getting frustrated with client.

 **Jonathan Aulson** 29:00  
And.

 **Andrew Scheuer** 29:09  
Because of the MCPS and it was, it was switching between plan mode, act mode, kind of Willy nilly and I was just consistency was an issue. Like one day it would be great and then the next day I would be having a lot of I would be struggling with it.  
And it would be more of an impediment. So I'm just, I'm trying out some new things and I saw a lot of really cool things with Claude, some really, really cool like agentic workflows that people have.  
Like even like I I said the cursor team in the chat said the cursor team uses Claude, which is funny because that's their competition. But but there's a lot of really cool flows that being being in the terminal for me that I didn't I didn't realize was was easier.

 **Jonathan Aulson** 29:44  
E.  
Hmm.

 **Andrew Scheuer** 29:56  
And a lot of integrations that can be done a little bit easier than using an extension which the that kind of has to the door has to be kind of opened for that like the the team has to has to make that available.  
Um.  
With like for like the data versus that was, I was that was a point of frustration for me, so I wanted to kind of solve that.

 **Johnn Hesseltine** 30:21  
Just wonder if there's is it? Are there just use cases right where maybe you go back and forth between the two? I don't. I don't know necessarily, right? But I just, I just liken it to.

 **Jonathan Aulson** 30:23  
It.

 **Andrew Scheuer** 30:29  
Yeah.

 **Johnn Hesseltine** 30:35  
I mean just like So what am what am I trying to say? So for example, so it's a level of abstraction like client gives you a level of abstraction.  
Above the command line level, similar to the way Power Automate is a level of abstraction over logic apps. Power automates logic apps in the end, right? But it's.  
But it's abstracted and.  
And equipped and armed with a more citizen developer friendly UI that. And by the way, like a lot of things that get abstracted, it's not quite as powerful, right? There's some limitations that you don't see.  
Say in logic apps, but what's the so logic apps? Less abstraction, more power, but requires uh.  
Kind of a different developer to operate in that area. So similarly I just and and by the way there are use cases for both, right? We use both. We have logic apps in places and instead use Power app or sorry Power Automate and others. Is that a similar conversation around client versus?  
Cloud for example in this in this context right where perhaps client is appropriate for call it less complex use cases where if there are more more complex requirements like.  
Uh, data verse work? Well then it sounds like Clyde may not be the the tool for that job, and in fact it's Clyde. Um, does that resonate guys? Or am I just like is that just way off?

 **Andrew Scheuer** 32:18  
Yeah, no, I think that if you know when you're when you're planning out a new project, I think looking at the tool, the A I tools that would be most appropriate to the to the project is is helpful. But but yeah, be be just I think being flexible especially right now cause it's it's changing every month.  
Um.

 **Johnn Hesseltine** 32:35  
Well, I mean, I'm even, I'm not even so certain it's at the project level, right? Is it could be at the, I mean it could be at the task level, you know?

 **Andrew Scheuer** 32:40  
Mm-hmm.  
Yeah, maybe.

 **Johnn Hesseltine** 32:47  
Where you may go back and forth between from task to task, you may be using different tooling in the IDE, right?

 **Andrew Scheuer** 32:56  
Yeah, I mean I've used three different ones in some features like I use client and then I use cursor and then I used Ali bot. So like you know, trying to just find the the the most appropriate tool for for the task.

 **Johnn Hesseltine** 32:59  
Yeah.  
I mean, how fluent do we feel like this team is and understanding which use cases?  
Or which tool I should say is appropriate for which use case, right? Like and I think that's the thing that I'd like to explore or or explore, but like.

 **Jonathan Aulson** 33:24  
8.

 **Johnn Hesseltine** 33:33  
Kind of true up and close that gap. If there's a gap there, I get the sense that there is. Maybe there isn't. But like if there's if there's a gap to be closed, I wonder if that's the one right get having. Yeah, some of it is you got to learn how to use these tools, but some of it is making sure you're clear about and familiar with and and fully understand which tool is appropriate for which.  
Scenarios and use cases. Um.  
And we should work together to figure that out, right? Like I don't, I don't know if anyone it's one person that needs to go to find that necessarily. But I think, I don't know, in some way it feels like it almost like a it's just a definition of not definition of dumb, but it's just kind of like a how we work type.  
Agreement amongst the members of this team and building up knowledge within this team to put opinions.  
Forth around that idea.  
And maybe we extrapolate that firm wide later, but.

 **Jonathan Aulson** 34:32  
Yeah.

 **Javier Casas** 34:33  
Andrew, correct me.

 **Jonathan Aulson** 34:33  
Um.

 **Johnn Hesseltine** 34:33  
Sorry guys, I don't know if I'm I don't know if I'm taking us on paths you don't wanna go down John, so I'm just gonna stop.

 **Jonathan Aulson** 34:40  
Yeah, no, these these are the paths. I I we do have to move on because we have a few more topics, but so I wanted to show a few things. There are two tools that I came across that are that I think are neat.

 **Johnn Hesseltine** 34:44  
Yep.

 **Jonathan Aulson** 34:57  
And I took the lazy way here. Unlike Andrew, I didn't make a presentation, but I have a YouTube video that we can look at. What I'm not sure of is if the audio is gonna come across so.  
I thought I would try.  
Can you guys hear talking just now?

 **Johnn Hesseltine** 35:23  
Hear you talking.

 **Jonathan Aulson** 35:25  
Let me, I'm gonna hit play again on the video and I wanna see if the audio comes through 'cause I'm not sure it will.

 **Andrew Scheuer** 35:25  
Yeah.

 **Graham Olson** 35:32  
I think there's a way if you share your screen, I think there's a drop down that appears on the right hand side possibly and then you can share.

 **Andrew Scheuer** 35:35  
Yeah, you can.  
On the on the top right, that's includes sound, yeah.

 **Jonathan Aulson** 35:40  
Oh, really?  
OK, all right, let me try this.

 **Graham Olson** 35:45  
Yeah.

 **Jonathan Aulson** 35:48  
Share sound. There we go. OK, all right. So this is an 18 minute video. I think I'm only gonna show like the first few minutes, but it's a tool called Warp. Yesterday I was sitting with a friend of mine who's a senior software engineer and watching him write.  
Is the audio coming through?

 **Johnn Hesseltine** 36:08  
Yeah.

 **Graham Olson** 36:09  
Yeah.

 **Jonathan Aulson** 36:09  
It is OK code for his new startup. Well, to be honest with you, I was really more so listening to him. That's right. He had multiple AI models running and he was literally just speaking to them and telling them very clearly what he wanted to get done. Now I'm talking about minute long, super detailed audio messages to these LLM's.

 **Andrew Scheuer** 36:10  
Yeah.

 **Jonathan Aulson** 36:28  
And truthfully, when I code now, I do a pretty similar thing. So sorry, just as context, this guy Tim is someone I follow. He's a senior engineer at Microsoft. Just as context, you're getting so good that nearly 90% of the code that I write is AI generated and I rarely need to write tons.  
Of lines manually, like I would have just 18 months ago. Now this means that integrating AI in your workflow is becoming more and more important, and if you want to keep up with those that are using it effectively, you really need to take this seriously. Now I say this to drive the point home that coding by hand is slowly dying. We're moving much more to a code by.  
Prompt world, and in today's video I'm going to prove that to you by demoing a very powerful tool called WARP. Now at a high level, WARP is an agentic development environment presented in an application that looks very similar to your terminal. It has a lot of very interesting features and it's bringing a new approach to agentic development compared to something.  
Like.  
Cursor or Cloudcode. Now I teamed up with them for this video and I'm going to give you a tutorial on how to take advantage of this tool, so stick around. All right, so I'm on the computer now and I want to go ahead and dive in. Now I'm just on the Warp landing page. You can see that I'm on warp.dev. I'll also leave this link in the description in case you want to check it out. And that is because if you want to use Warp, you do need to.  
Download this tool because it is a standalone application. It's not for example a CLI tool like something like Cloud Code. OK, so go ahead, download this if you want to mess around with it. It is free to download and you can do this for your respective operating system or you can use something like Brew to install it.  
Now, when it comes to the price, there is a free tier, of course, so that's the one you can mess around with. And then if you want, you can upgrade to the Pro Turbo Enterprise tiers as well. Anyways, let's hop over to the editor because I want to show you what this actually looks like. OK, so this is Warped. I opened it up. Yeah, it's a standalone application, so you've got to install it first. And from here, this probably looks pretty familiar.  
Because it's very similar to the terminal environment that most of us are used to typing in. Now I wanna show you a few things that this can do and then I wanna talk about the logic behind this user interface and how this compares to let's call it like a full-fledged kind of IDE, something like cursor where 80% of the screen it's like files and code and all of that kind of stuff.  
OK, so from here I can use this like a normal terminal. So I can do something, you know, mktertest, right? And I can just go ahead and create that. Or I can actually ask in natural language. So I can say change to my desktop. OK, so something quite simple, but it will actually detect, all right, they're using natural language here. It's not trying to run a command.  
So let me generate the correct command. OK, boom, let's run that and then it will change us to the desktop. OK, so we're now in the desktop directory. You can see that here. Now from here I can say mkdir test OK like that. So I've made the directory. I can CD into the directory and I can just use this like a normal terminal. But then where it becomes quite powerful is that at any point in time.  
I can use the AI integration. So here I'm gonna use LS. We see that we don't have anything and I'm gonna say make me a new Python file to print a Christmas tree or something. OK, Christmas tree like that. So I should go ahead and put that inside of this folder. So let's see what it does here.  
See that it's loading a diff and then it should be able to generate the code. OK, and there we go. You can see that it's now generated this Python script for me. Pretty basic. So you guys get the idea. I think natural language processing in the terminal there is.  
Uh.  
I'm gonna try and warp ahead here. Uh.

 **Andrew Scheuer** 40:03  
That's a good one.

 **Jonathan Aulson** 40:04  
Yeah.

 **Javier Casas** 40:07  
Well done, John.

 **Jonathan Aulson** 40:09  
Yeah, I stay up all night just thinking about these things, you know?

 **Andrew Scheuer** 40:09  
Yeah.  
Yeah.

 **Jonathan Aulson** 40:14  
Uh, let's see.

 **Johnn Hesseltine** 40:16  
Did you get the return on that investment just then?

 **Jonathan Aulson** 40:18  
Yeah, it's it's, you know, it's it's compounded interest. It builds up overtime, Tom.

 **Andrew Scheuer** 40:25  
This is why This is why it wants a larger audience.

 **Johnn Hesseltine** 40:25  
Got it understood.

 **Jonathan Aulson** 40:27  
Yeah.

 **Johnn Hesseltine** 40:27  
Yeah.

 **Graham Olson** 40:27  
I I laughed, but I was on mute. I just want you to know that.

 **Jonathan Aulson** 40:31  
Just just wait till my till two more presentations. I got a good one. All right. Well you know what with this one I'm I'm gonna stop it here cuz like this is very similar to actually the Claude code. I didn't know Andrew that.

 **Johnn Hesseltine** 40:36  
I.

 **Jonathan Aulson** 40:50  
I didn't actually hear about cloud code until I was watching this video, so I didn't know we were gonna present that anyway. So this is just another I think tool. It seems interesting for but I'll link this video. I'll do that cuz there is a lot more information in it. I don't like see the section I'm trying to.  
to find, but there's some interesting scalability implications with this tool. So let me link this.

 **Andrew Scheuer** 41:18  
Yeah.  
Yeah, well, what got me... Sorry, Rim. Sorry. Go ahead.

 **Graham Olson** 41:21  
Do you like that is?  
You can go ahead.

 **Andrew Scheuer** 41:27  
Well, no. What got me in my ears perked is the the idea of having like he said something about agents. I don't know what exactly he said. And there's a lot of people starting to work with.

 **Jonathan Aulson** 41:40  
Yeah.

 **Andrew Scheuer** 41:44  
Like multiple LLMS at once and coordinating them between each other. I was trying to play around this weekend with creating a a few like a whole dev team with agents and having them communicate and discord to each other.

 **Jonathan Aulson** 41:46  
Um.  
Yep.  
Oh, interesting.

 **Andrew Scheuer** 42:01  
But I'm afraid that the matrix could explode when I do that. But but it's it's an interesting over complication of what what I'm of what I'm seeing a lot of other people do. So I I want to look into warp more. The difference between warp and cloud code just from that what I saw there is cloud code can only work in specific.  
Directory. So you can't like if you try to like go out and change directories, like go to a different directory it like has. I think they block it for probably very good reasons, but yeah, it doesn't let you. So warped it feels like it's like an AI agent on your desktop type of thing.

 **Jonathan Aulson** 42:21  
Mm.  
Yeah, yeah, yeah. And that's that's the part of the video that I I didn't see with a quick scan is he goes through and and has several agents working in concert, which is interesting.

 **Andrew Scheuer** 42:38  
So.

 **Jonathan Aulson** 42:53  
All right, so then the other the other thing I wanted to show is a system called Pythagora. This is a new all-in-one coding, prototyping and deployment just like Vercel, right? So our vo.dev.  
What I really like about this one is that it has a very structured process to to to the process. So and what do I mean by that? So when you start the first prompt you give it will.  
Will create a spec and so it starts it starts iterating with you on the spec before it ever does any kind of prototype or UI, right? So it goes through these phases of spec task.  
And then code, debug and deploy. So it's A and at each step it assumes you're trying to move into production. So like with VO dot dev right when I when I prompt something.  
That has to do with any kind of data. For example, it creates mock data within the page that within the feature itself, the component that I'm working on right as opposed to creating a table.  
In a database and and seeding data there and having the connection pull it up through the UI. So from like the very start it's actually building a production system as opposed to a mock up, right? So that's that's the real critical.  
Difference that I see with this one is that you can you can move through a very structured process of specs, task breakdown structure to like task everything out, make sure that that all makes sense and then working through a prototype and deployment and there's a debug.  
Phase as well, where it actually goes through the logs with you. Andrew, what's up?

 **Andrew Scheuer** 44:59  
No, I I I just the having the database part is interesting considering the the the difficulties that we had with a couple stories ago with not having a good clear understanding of the data structure.

 **Jonathan Aulson** 44:59  
And.

 **Andrew Scheuer** 45:16  
And that caused a lot of like cause a lot of wrinkles in the in our development process. So maybe it might be a good, I don't know tip to to implement that or design that first.

 **Jonathan Aulson** 45:16  
Mhm.  
Yeah, now I I that's what I want to do with this is if we can get our our entire schema into a mockup here, then we can be, you know, we we can be creating features.  
That match our back end, which I think is basically what you just said as well. So anyway, so this this is a a cool thing I was excited about. I'm going to keep playing around with it. So far it looks oh and the tech stack is. It's a node dot JS back end. It is.  
React front end it it starts off with a full a WS infrastructure but it it's agnostic. You can you can you can change that over to a different provider. So presumably we could use Azure with this.  
Although you have to, I think you have to like, well, I haven't really gotten to that part of it yet, so I don't really understand that piece. But anyway, so that's that's that one. Oh, and the database, I think it starts with its default is Mongo DB, but.  
I had. I haven't really looked into what options they have there. Anyway, that's that one. And then I guess, does anyone have like questions or anything on that one before I move on?  
Yeah. OK. So the yeah, yeah.

 **Javier Casas** 46:57  
Yeah, I have one. Sorry, can we can we load?  
Uh, repository on this or you have to start from scratch?

 **Jonathan Aulson** 47:09  
So far I've only seen an option to start from scratch, but you can iterate within the project you start. So I assume we could like craft a prompt that essentially implemented.  
A. A schema that's similar, you know.  
Yeah.

 **Javier Casas** 47:30  
Nice.

 **Jonathan Aulson** 47:32  
But yeah, I need to look into what options they have for that. OK, so this next thing is an idea I have for an internal tool and I did some work over the weekend.  
Trying to flesh it out. Unfortunately the tablet I did my work on is there's some kind of issue. The router won't let it connect so and I haven't finished troubleshooting it, but so I took a photo.  
Of a drawing. So this is John. I know you're going to be tempted to give me some kind of architecture award for this, but you know, hold on until the end.

 **Johnn Hesseltine** 48:20  
Yeah, don't worry.

 **Jonathan Aulson** 48:20  
So yeah, yeah, let me zoom in on this. So here's what this is. I face a issue with our data product trying to get data into the product, right? It's challenging because you have all these different sources.  
Of data and you want to you want to take those things like meeting transcripts and ingest them into a a markdown format that's organized differently and you want to you want to tag it with metadata and you want to file it away into the data product.  
So there's several steps to this, right? And so I was thinking, wow, it would be great if I could automate that. And I was showing John on Friday. I created a a copilot agent to do this and it failed miserably. It hallucinated all over the place. It was completely useless. But so I I kind of went back to the drawing board.  
And the current version I have is I'm going to talk about the back end first. So it is a Docker container with with essentially running a fast API API that does a few things so it identifies the source.  
Of the document, it identifies the what type of document it it's it is. So like a meeting transcript is the example. That's what I have as MVP right now. So it identifies the fact that it's a transcript.  
And then it sends a set of templates and a prompt to an AI API, right? Right now I have it using Open AI, but presumably we could plug this into chat to a later bot.  
So the the A I model receives a YAML for the document, a a document specific markdown template, right? So think like outline organization of meet for the meeting transcript information.  
And then a prompt based on the the type of document. And so then the the AI agent turns that into a, you know, rich metadata, organized information markdown document.  
Tagged with metadata that's specific to your project and and document type, right? And then that agent will save it using SharePoint's APIs. Save it to your data product folder in SharePoint.  
So that's what that's the purpose of the tool, right? And then the usability is kind of on the front end. So the front end I I'm picturing 4 different entry points. I've built out three of them, so there's a teams bot.  
A chat bot in Teams that will accept a command to upload and so it's upload and you attach the document and that will trigger the bot to send the document into the API. There's a Power Automate flow I have that has an e-mail trigger with the flow.  
Specific e-mail address. So if you wanted, you could just attach the document to an e-mail and send it to this specific address and the flow will strip the document and send it to the API, right? It'll also take a curl command from a terminal to do this, to move the document into it, and then the.  
4th one would be a tool within a data bot right? So that you could tell a data bot hey grab this meeting transcript for my Microsoft 365 and process it and put it in my data product right? Or presumably we could even have something like for meeting.  
Transcripts specifically where every time a new one shows up and I checked this out, the Microsoft Graph API allows for this scenario. It'll pick up the meeting transcript off the meeting, process it and put it in SharePoint for you automatically.  
So.  
So that's my pitch. Um, what do you guys think?

 **Andrew Scheuer** 52:39  
Sorry, I'm I'm muted. I like it.

 **Johnn Hesseltine** 52:41  
Oh, sorry, I was also muted.

 **Andrew Scheuer** 52:46  
Like it.

 **Johnn Hesseltine** 52:46  
Um, John, you're responsible for drawing, drawing this diagram.

 **Jonathan Aulson** 52:51  
Look, I have a better one in Mermaid on my tablet, which I can send you as soon as it gets reconnected. Yeah, yeah, yeah.

 **Andrew Scheuer** 52:54  
Quite good.

 **Johnn Hesseltine** 52:58  
Yeah.  
I can't wait. I can't wait. No, I I love it. I think. Do you? Are you able to demonstrate this right now?

 **Jonathan Aulson** 53:10  
So I'd like to ask for volunteers because here's what I have. Let me show you what I have. I have this set of code. So and this is I think I I think I saved this to a repo in GitHub. I can't remember if I did the.  
Latest version or not, but it will.  
So the teams bot is, first let me say 100% of this code is AI generated. Olson did not touch any of this code, so that should be reassuring. Yeah, but obviously because it's AI generated, we're gonna, we're gonna, it's gonna need some work, right?

 **Johnn Hesseltine** 53:47  
Yes.  
Yes.

 **Jonathan Aulson** 53:57  
But I've got a Teams bot presumably developed with the deployment instructions and and images as well, right? So this this is one part of it. I've got the Power Automate flow in Jason.  
And then a setup guide for it. I've got the API along with the prompt and markdown templates and YAML for meeting transcripts, which is the first document type I want to go after.  
And a deployment guide for the Docker container. I was thinking it would run on Azure App Services, but you guys tell me right anyway.  
And the AI plug in, I have like the idea written down basically and how it should work. So that's kind of where the state each part of this is in. I'd love to like move it to the next level of proof of concept, but I need some help to do that.

 **Cesar Figueroa** 55:07  
Uh, what kind of help do you need? Like to deploy testing? What part do you need?

 **Jonathan Aulson** 55:13  
Yeah, yeah I yeah I need. I need some help getting a proof of concept up and running. So I need like the Azure if it's app services right? I need I need one of those things. I need a docker container and I need this this.

 **Cesar Figueroa** 55:14  
All that.

 **Jonathan Aulson** 55:31  
This API script deployed within it. I'm only like really kind of hazy about what I'm talking about right now, so I'll need some help kind of figuring out what I don't know, but.  
But that's kind of the idea.

 **Cesar Figueroa** 55:51  
Um.  
Well, now that Javi is here, I think that the team can focus on spring working and that means that I don't have too much features in the spring to work on.

 **Jonathan Aulson** 56:01  
Mm.

 **Cesar Figueroa** 56:07  
So if you want, I can spend some time reviewing what you did and helping you view like this in your data Azure account.

 **Jonathan Aulson** 56:15  
Yeah, that'd be awesome. I think we need to time box it, right? Because I don't want to like my hobby projects to take up too much of our work time. But yeah, that'd be great, man. If we could get that going, I will reach out.  
Find some time with you. Does anyone have any like comments or ideas like oh hey why you know doing it that way doesn't make sense? Or there's you know anything like that stand out? Like am I spouting nonsense when I say API within a Docker container on Azure App Services or does that make sense?

 **Johnn Hesseltine** 56:51  
I'm just not. I'm just not convinced Docker's required. That was my only thought there. I don't think you have to and Python like. I don't know why if Python was just.

 **Jonathan Aulson** 56:55  
OK.

 **Johnn Hesseltine** 57:06  
AI drove you to Python, John, or if there was some or there there was some intentional deliberate choice there. Um.  
But to me like.  
Some you know TypeScript or JavaScript running on some sort of uh.  
Hell, well, you know, it may not even, it may not even be an app service by the way to potentially even be an Azure function. OK, I I, you know, I need to, I need to maybe fully or, you know, start to drill in a little bit more. But that was my only comment like Docker and Python. I'm not convinced that's.

 **Cesar Figueroa** 57:37  
Yeah.

 **Johnn Hesseltine** 57:45  
Necessary but.

 **Jonathan Aulson** 57:47  
OK.

 **Johnn Hesseltine** 57:49  
But otherwise, I mean that's that's more just implementation specifics. I mean the broader strokes of the.  
Concept.  
I think it makes sense the YAML part. I don't know if I followed fully. Uh John, if you wanna go back to the.  
Diagram for a second.

 **Jonathan Aulson** 58:09  
Yeah.

 **Johnn Hesseltine** 58:11  
I wanna make sure I got because I I what I read here is send to AI YAML front matter type specific. So I think this is the part I I don't know actually tell me more about that that bubble and the and the words next to it.

 **Jonathan Aulson** 58:12  
Good.  
Sure. So the API is responsible for identifying the source because oh, I did. I don't think I talked about this. My my intent with this is to have a single back end.  
That's able to serve all projects, all documents. In other words, we would we would have kind of, we would be able to centralize the logic involved with.  
Choosing well centralize the logic I think where it makes sense. So like the logic here in the in the API here. Um.  
No, it it based on where, based on what it knows about the user, right? So if it's a Teams chat bot, it knows what project you're in and what and so it can associate you with certain templates, right?  
And so it will it will pick the template and identify the document type, which I don't have that part implemented, but that's the theory. So like the logic for doing this stuff lives here in one place source document type and then picking the templates and prompts that go with those things.  
Right. And then the logic for how do we, how do we assign metadata and the actual interpretation of the document is all within one AI API. So what I've tried to do here is like.  
Centralize and segregate the the lot, the chunks of logic so that we would have like maintainability going forward across the whole firm. What is the goal?  
I'm sorry. And so you were talking about the YAML. So the YAML is intended to help the A I with context for the for interpreting the interpreting the document.

 **Johnn Hesseltine** 1:00:23  
Is sure.

 **Jonathan Aulson** 1:00:39  
And a template.

 **Cesar Figueroa** 1:00:43  
Like the rules for example.

 **Jonathan Aulson** 1:00:44  
OK.  
Um.  
I'm so my understanding of YAML is that it kinda can help set the the the the set of specific metadata tags.  
That might be used to then tag the document in the template. Is that does that align or does that check out John?

 **Johnn Hesseltine** 1:01:19  
Hello.

 **Jonathan Aulson** 1:01:21  
I did close to 90 seconds of reading on this, so I'd be surprised if it doesn't, yeah.

 **Johnn Hesseltine** 1:01:24  
Yeah, yeah. I mean, Yam. Well, Yam will.  
Is effectively.  
The data. It's a data structure in the end. Not so dissimilar, say from Jason, frankly, but using a different.

 **Jonathan Aulson** 1:01:45  
Oh, really?

 **Johnn Hesseltine** 1:01:49  
Syntax, I guess you could call it in the end. That's the way I think about YAML. Um.

 **Jonathan Aulson** 1:01:51  
Yeah.  
OK.

 **Johnn Hesseltine** 1:01:57  
You seemed to. Oh, I don't know. You seem to describe more.  
You seem to to interpret YAML as like.  
I think like almost as if you were building business logic into it or logic, you know, business logic with some level of logic. I'm not sure. And maybe I misinterpreted if that's not to what you were doing, but like it's a it's a data structure like a Jason object, just format it differently. That's that's the way to think about it and that doesn't doesn't feel like that.  
That aligned to what you were saying though.

 **Jonathan Aulson** 1:02:31  
I I think it it might, although I didn't realize it was an entire like like similar to Jason. I thought it was specific to metadata, but maybe there's not. Maybe that's blurring anyways, so the.  
My understanding of the purpose for this piece in the process is that it would have a list of project specific metadata tags and the AI seeing that and then getting the prompt and the and the content of the document along with a template.  
To put it into would use the YAML as a reference and say, OK, you know, for this, for this document type and this project, here's the list of metadata that they would use and here's here's the here's the association of metadata to data that they would use to kind of map that.  
So that ultimately you get saved in the right place or with the right data and in the right place.

 **Johnn Hesseltine** 1:03:31  
Yeah, I I mean, you could probably use it for that, but I don't know.

 **Jonathan Aulson** 1:03:36  
Yeah.

 **Johnn Hesseltine** 1:03:39  
That that's all I that may not be the thrust so much of this conversation, but is that AI bubble?

 **Jonathan Aulson** 1:03:44  
Yeah, OK.

 **Johnn Hesseltine** 1:03:52  
Is that really just code that's running that's connecting to LLMS?

 **Jonathan Aulson** 1:04:00  
Yeah, so I have a open AI plan that allows me API access and so that's the end point that I would use for, well, any of the any of the open AI models. I assume we have a similar thing.

 **Johnn Hesseltine** 1:04:07  
Yeah.  
Sure.

 **Jonathan Aulson** 1:04:17  
That could be used with a later bot under its different models, right? So.

 **Johnn Hesseltine** 1:04:21  
Yeah.  
Yeah, OK.

 **Jonathan Aulson** 1:04:25  
Yeah.  
All right guys, well, I've run us over. I appreciate the feedback and the offer for help. Anyone have any other items?

 **Johnn Hesseltine** 1:04:40  
Um.  
I just well, I don't know. I don't know if this is for this group or John, this is just more for you and me.  
Maybe this is just you and me, but we can.  
If you just have a couple extra minutes for just you and me, that would be great.

 **Jonathan Aulson** 1:05:02  
Sure. Chris, I think I have a call with you, Sir. I'll ping you when when we're done here.

 **Christopher Thompson** 1:05:08  
OK, sounds good.

 **Johnn Hesseltine** 1:05:09  
It 2 minutes. That's I don't need it a lot of time.

 **Jonathan Aulson** 1:05:14  
Thanks guys.

 **Christopher Thompson** 1:05:16  
Thanks everybody.

 **Johnn Hesseltine** 1:05:16  
Alright, see you guys.

 **Jonathan Aulson** 1:05:18  
Thank you.

 **Johnn Hesseltine** stopped transcription