Katie (00:00:02):

All right, we're live.

Damian (00:00:05):

Sweet. Excellent.

Katie (00:00:13):

We're up to 18, but two of them are internal.

Damian (00:00:16):

That's okay. That's good. Who can share? So, I'm just going to see if I can share my screen as well. You may need to give me permission for that as well.

Katie (00:00:32):

I think I just have to stop sharing.

Damian (00:00:35):

Oh, okay. Yeah, that's right. We should be fine.

Katie (00:00:41):

I'll stop it a minute before, but-

Damian (00:00:43):

Yeah. Then, I'll switch over. Sounds good. You got five minutes.

Katie (00:00:48):

Hey, Andrew.

Damian (00:00:53):

So, did you get my email this morning?

Katie (00:00:56):

Yeah.

Damian (00:00:57):

The latest deck. Yeah, cool. I'm going to see if I can run the slide deck off of my phone.

Katie (00:01:07):

Ooh, fancy.

Damian (00:01:09):

Yeah, apparently, I did it once before and it was good for the first half and then it decided to die on it. So, I got through half the slide deck and then I went to manual. How's the weather out there in sunny Alabama?

Katie (00:01:33):

Good.

Damian (00:01:34):

It looks sunny.

Katie (00:01:35):

A little too bright. There we go, that's better. It's good. I think it's in the 40s today.

Damian (00:01:43):

What's the climate like in the winter?

Katie (00:01:47):

Well, right about now, typically when we have maybe our last heatwave of the year... But, we haven't had that so far, I guess, because... I don't know.

Damian (00:02:05):

Yeah.

Katie (00:02:05):

But, it's humid and it's cold, but not in the 20s cold.

Damian (00:02:09):

Yeah, no, that's cool. Yeah. My one visit there, it was summer, so it was pretty humid.

Katie (00:02:18):

Yeah.

Damian (00:02:19):

I do remember that. But, it wasn't horribly uncomfortable. It was not as bad as some humid. I don't know how hot and humid it can get in the summer, but it wasn't too horrible.

Katie (00:02:33):

It can get pretty toasty.

Damian (00:02:34):

Yes.

Damian (00:02:41):

So, have you got plans for the holiday period? Are you hunkering down?

Katie (00:02:46):

Yeah, just hunkering down.

Damian (00:02:48):

Yeah.

Katie (00:02:48):

What about you?

Damian (00:02:53):

Same. Same thing. Occasionally, in the past, we've had family members come and visit. My family's scattered all over the place. Some of them obviously back in the UK. My brother lives in Perth in Australia with his family. My other brother lives in France, in the Pyrenees.

Katie (00:03:18):

Wow.

Damian (00:03:18):

My sister's back in the UK. So, yeah, normally Christmas is just the immediate family and then a few friends. So, it's not going to be too dissimilar to normal Christmas' for us, I don't think. So, where did you find this clock? It's great. Yeah.

Katie (00:03:49):

What are everyone else's holiday plans? You can throw it in the chat, and-

Damian (00:03:55):

Yeah, Q&A. Yeah, see if there's some interesting ideas under the current circumstances. I'm going to tell you what we have found fun, is Jackbox Games. You've ever come across Jackbox Games?

Katie (00:04:16):

Yeah.

Damian (00:04:17):

Yeah. Yeah, they're pretty fun and family friendly. Everyone get on their phones, their devices, and play. It's kind of, fun. It's a good social activity.

Katie (00:04:33):

Yeah, no, it's fun. I actually hosted a couple happy hours and use the Jackbox Game platform to like-

Damian (00:04:47):

Yeah.

Katie (00:04:48):

Yeah.

Damian (00:04:51):

Yeah. No, that's brilliant. I'm just going to get a quick top of water and then I'll be back two seconds.

Katie (00:04:59):

Perfect. Hi, everyone, thanks for joining. We will get started here in about a minute according to the timer. Damian just went to go get some water and we will get started here shortly. Hi, thanks for joining. Damien's back.

Damian (00:05:38):

I am back. Everyone being Agile-ists, of course, they're all going to be on time. There won't be any late-comers, right?

Katie (00:05:57):

Fingers crossed. Although, I still figure, we'll start a couple minutes late.

Damian (00:06:01):

It's a key principle.

Katie (00:06:17):

All right, Damian, I'm going to go ahead and throw it over to you to share your screen.

Damian (00:06:21):

Okay, I'll do that.

Katie (00:06:25):

You can see that you have some fans in the chat already coming to say hey.

Damian (00:06:32):

Excellent. Yeah, there's some good 'ole friends. A lot of St. Louis participation today. That's fantastic. Good to see all you guys. How's everything going? It's been a crazy year in lots of different ways. But, yeah, we're closing in on the end of it, so look forward to 2021 and how we deal with that. But, good to see everyone.

Katie (00:07:02):

We are so happy everyone's here.

Damian (00:07:07):

Can you see my screen, Katie?

Katie (00:07:09):

Yes. I can see your screen.

Damian (00:07:16):
I can hear a cat in the background.

Katie (00:07:19):
Yeah.

Damian (00:07:20):
Is that one of yours?

Katie (00:07:24):
Yes, she's bringing me a toy.

Damian (00:07:27):
Well, she needs attention.

Katie (00:07:35):
I knew shutting the door and locking them out wouldn't work.

Damian (00:07:39):
Yeah. They'll get very frustrated with that. Believe you me, I know exactly how that is.

Katie (00:07:45):
Yeah. All right. Let's see, it is one after. We'll get started at two after, I think, just to give people a little bit more time to join and get their notebook and their water. Sorry about the cat meowing in the background, that is just my life.

Damian (00:08:11):
It's good background sound.

Katie (00:08:14):
Damien also has cats.

Damian (00:08:16):
We do have cats-

Katie (00:08:17):
He understands my plight.

Damian (00:08:19):
Yep.

Katie (00:08:21):

Hey, Doug.

Damian (00:08:24):

Completely understand.

Katie (00:08:27):

All right, well, looks like it's two after so I'll go ahead and kick everything off. So, good morning, good afternoon and Good evening, everyone. I am so pleased to see such a wonderful representation of our community here on this webinar today. So, thank you for taking the time to join us. My name is Katie Frank, and I'm the content marketing manager here at Veracity. And, I'll be your host and moderator for today's webinar. If you're not familiar with Veracity, we help enterprises, especially those in highly regulated industries with custom application development and adopting Cloud native technologies as a platform for innovation. We plan, build and launch successful software solutions from conception through adoption.

Katie (00:09:14):

Our expert advice, coaching and skilled developers can help your organization identify and achieve critical business goals and lift your team in the process. I would love to encourage you to submit your questions for Damian through the Zoom Q&A interface. So, just hover your mouse over your Zoom window and you'll see a little button that says Q&A. Damian will answer your questions at the end of his presentation. Also, feel free to submit messages in the chat. I'll be moderating that and can pop up with any random questions that come up or anything like that. Again, thank you for joining us today on this webinar and On Demand. Allow me to introduce today's speaker, Damian Dingley.

Damian (00:09:59):

Thank you, Katie, very much. Welcome, everybody. As I just mentioned earlier, good to see a few familiar faces, familiar names, at least, out there. So hi, everybody, hope things are all going well for everyone. All right, let's tackle this. Before we do, for those who don't know me, or not so familiar, a little bit of background here. So, I've been working with Veracity for about seven, eight years, and in the Agile space, probably 10 years. Over the course of that time I've worked with multiple companies, both big and small. You can see a little subset of the organizations I've been involved with, with some relative size there. So, all quite different experiences, all with a common goal of moving, either picking up Agile from the get-go or readjusting. Resetting things if Agile's got a little derailed. All been really fascinating experiences.

Damian (00:11:05):

Today's topic on Agile tooling is probably the one that generates the most interest, emotive, kind of, controversial sometimes, it can be. So, we're going to look at that today and look at some of the nuances, some of the characteristics, of the tools that we're looking for. What are the things we want to look out for in picking up tools of choice for Agile. So, that's what we're going to look at today. We've actually created two webinars. This first one's going to look at the go-to's. The next webinar in January is going to look at... We're going to widen the scope a little bit and look at some of the other tools, especially around the remote distributed teams and things like that, which is particularly pertinent right now under our COVID situation that we have across the globe.

Damian (00:12:02):

So, that's a little bit about me, let's dig in. How much is out there? How many tools are there out there for us to choose from? Well, it turns out, there are hundreds of them. Back in the day, there was a couple here and there. But, not much automation, not much in the digital space. But, there is there's a good chunk of tools out there for us to pick from, and they all serve different purposes. So, this slide is really kind of... This is my tree of tools, the tool set. This slide is indicating a sample of companies and organizations that provide tooling for the Agile space, the Agile platform. You can break them down into those categories I've identified on the left there on this bullet point.

Damian (00:12:53):

So, your classic backlog organization through to enterprise management, through the knowledge distribution, distributing knowledge across teams, a whole bunch of stuff. So, it's a big space. We could probably have a whole series of webinars on each one of these topics. We're going to tackle this into two stages, as I mentioned earlier. But, there is a lot out there. So, how do we get through all of that and try and identify, maybe, the key ones, at least from my perspective that would that are important to spend a little time understanding, digesting.

Damian (00:13:36):

So, what better than to go to Gartner and look at their reports, and see what Gartner says about a lot of this stuff. So, for those familiar with Gartner, they produce annual reports on a whole bunch of topics, and they have a whole section dedicated to Agile and, in particular, Enterprise Agile tools. So, you'll see from this quadrant here, the magic quadrant, who the lead players are, who's up and coming, some of those niche players. So, I always want to put this talk together. I went back to Gartner and said, what's the domain looking like? What's the market looking like for these different organizations? Who's using what tools, what's the more popular ones?

Damian (00:14:25):

Of course, if you look at the leaders quadrant there in the top-right, you see all the... The classics are there, what you would almost expect. I don't think there's too many surprises there. Maybe, Planview possibly. But, you see Atlassian their big time. Broadcom, which is Rally, essentially and some of their associated tools. So, no huge surprises there. What was interesting to me there is, some of these niche players... Things like IBM, so they definitely... IBM have their own flavor of things. You can go back to Rational Rose even. So, they've specialized, and especially on the enterprise environments and ecosystems. They're big players there, and they do some really good stuff in that area. So, that's kind of interesting.

Damian (00:15:21):

The challengers, the top-left quadrant is kind of, interesting, too. There's Microsoft sitting there. It always brings a smile to my face when I see Microsoft. Because, Microsoft, their key to all our development tools. Very recently, with the community additions and open-source stuff they've kind of moved into, they've really expanded their reach, and are the go-to's for a lot of our development environments. They've always played a bit of catch up in the Agile world. Those who are familiar with Microsoft tools might remember their first go-around trying to catch into the Agile space. They came up with a few. They modified a few things in TFS, in Team Foundation Server. It wasn't that good. In fact, I was asked to look at it and I shied away quite quickly. That was a few years ago, they have come on leaps and bounds. And, I think with the onset of Azure DevOps now, and the evolution of TFS, you can see how they're coming up there in terms of challenges. But, they always seem to be ending up in that quadrant, they don't ever quite seem to get there, against the players like Atlassian and Broadcom and some of the others.

Damian (00:16:39):

Anyway, so that's a little bit about what Gardner's take is on things and identifying where the players are and where they sit. So, I took this as my starting point, and then decided to dive in a little bit deeper on those main key players. So, let's take a look. I broke these into categories. So, based on that set of bullet points we saw on the earlier slide, I said, "Okay, this is a big topic. How are we going to break this down? How are we going to tackle each one of these? So, I came up with these categories. You might disagree with them, but this is how I saw the big picture here. So, we have on the left, we have those go-to's. All right? Things that we need every day, the key to our Agile, day-to-day operations, if you like. Things that support our stand ups. Those kind of things, managing backlogs and things like that.

Damian (00:17:40):

Then, we have planning tools. And, we have tools that are dedicated just for planning, or we have add-on's or extensions in all the cases, to our go-to tools that provides the planning capability to us. So, we'll look at those. Then, there's distributed team. So, this is become more and more prevalent. For Agile and for software development, we've always had an element of remote teams, offshoring teams, teams in different locations. So, we've always been pretty cognizant and pretty aware of what happened. Things that we need to support our teams who aren't sitting face-to-face.

Damian (00:18:22):

Of course, in the last eight to 12 months, everyone's nearly a distributed team. In fact, I haven't even met Katie face-to-face yet, right. So, that's the situation we're in. But, it deserves some time looking at that. Actually, I find that this topic, that distributed teams topic, is extra fun. Because, there's some cool things out there in terms of remote office, and those kind of things that can really help us try and feel like we're collaborating, we're closer, we're organized. We can keep an organized service. So, there's some fact tools out there.

Damian (00:18:59):

Then, finish up knowledge share. So, important to make sure that the information about our development teams, and our product organization, product owners, they can distribute knowledge effectively across the team, across the organization. So, there's a section dedicated to that. Last, but by very means not least, is value, measuring value. Sometimes that gets brushed under the carpet a little bit. In the last, probably year, year and a half, especially around DevOps, measuring value has become more and more important, and I focused a lot on that. So, we'll tackle value again in part two of this.

Damian (00:19:46):

So, for today, we're going to dive a bit more deeply into the go-to's of the planning. That will be part one. Then, in January, we'll tackle those other three items.

Katie (00:19:58):

Hey, Damian.

Damian (00:19:58):

So, that said... Katie, yeah.

Katie (00:20:00):

We actually had a question submitted, and they're asking if you can give a quick definition of what an Agile tool is?

Damian (00:20:07):

A definition of an Agile tool. That's a really good question. An Agile tool. Well, I think, in some ways, as we go into this part one, that will become almost self-explanatory. But, to sum it up, it's a function, it's a capability that allows us, in the Agile space, whether we're acting as Agile coaches, scrum masters, or just part of a team, part of a development team, or in the product part of the organization, something that allows us to move the business value from an idea, through to development, through to something that's actually running this deployment as functional and is delivering value. So, it's those functions, those capabilities, that allow us to go from those capturing those ideas, managing that, in some form, working through that, what we call the backlog, through to delivering the value. It's the functionality that allows us to do that.

Damian (00:21:19):

Now, you're going to see if we go to the go-to's... As we go through this, it may become more self-explanatory. Those tools can be as simple as, [inaudible 00:21:28], what should we say, sticky notes. There's your tool. You could quite effectively run a whole Agile organization using sticky's and post-it's and whatnot. I'm not going to say that's the most efficient way of doing it, but that's the simplest tool we have in our arsenal of tools that could work effectively. There's good reason for it, too. So, you'll see here in this picture here that we're working on... I think this is a planning session that I did a couple of... This is when we could all get together around a table, look at that. Do you remember those days? Look at that. Sorry, it's fascinating.

Damian (00:22:15):

But, the idea here is that this very simple tool is so effective. One, you can see the focus on everyone's face, right? Number one. It is a way of gathering people together and focus on an idea, and it's very interactive. It's very tactile, because it's physical. I think everyone on this call, probably, who's done any kind of Agile, knows the sticky very well and knows how it can work so well. The challenge is, as you scale out, and things get bigger, and you got more teams involved, how do you transfer all that information to something that everyone can share easily? Sure, you can have walls stuck with sticky's all over the place. And, that serves to a certain purpose. But, how do you go beyond that? But, the simplest tool, I think, we have, and we should never forget it, is that sticky. Writing ideas down, creating backlog, understanding features, looking at risks, looking at dependencies, all those things can be captured in those sticky's. And, it's a great collaborative tool.

Damian (00:23:24):

Yeah, as you can see in those bullets, it engages people. So, that's always my go-to. I think whenever we're doing early planning work, inception work, or just trying to break down the work into smaller chunks as it were, the sticky's a great one. So hopefully, that starts to provide some explanation behind what the definition of what is an Agile tool. There's one right there, okay? And, it works very well.

Damian (00:23:54):

As I said, though, if you want to take it a little further, and especially if got distributed teams, and you scale things out, how are you going to transfer that information from this lowly little sticky, sticking up on a wall, to something that everyone can share across the whole organization, effectively. So, the first step would be a camera and a spreadsheet. Okay? Take pictures, and then start writing those up in a spreadsheet. Then, that spreadsheet can be distributed. So, that's your next level, if you like, beyond the physical sticky note.

Damian (00:24:34):

The next level beyond that, is a tool that is always a go-to for me, which is TRELLO, the digital sticky note. I think everyone's probably used TRELLO in one form or another. I think I use it for all kinds of things including retrospectives. Some of the people on the call today, probably, have been on my scrum teams... Probably are familiar with me pulling up TRELLO to capture feedback on a retrospective and to persist it, so people can look back at it. But, TRELLO for me is, simple, you can use it out of the box, there are additional add-on's and things, especially for Agile. But, you can use that tool out of the box. There's a free version of it. It's effective, you can transfer those sticky notes, if you like. Straight to that. You can create your progress lanes, backlog lanes, move those sticky's around in real-time and share that. It's a nice tool, and very effective, easy to use. So, again, that's my go-to, if we're keeping things relatively simple, but we want to move fast on something. So, lean and mean.

Damian (00:25:53):

Okay, how are we doing for time, Katie?

Katie (00:25:57):

We are doing good. It is 19 after the hour.

Damian (00:26:01):

Okay. Perfect. Okay, what do we have next? So, we're going to go... We're increasing our level of sophistication here. This is that next level on, its backlog management. So, if we're going to go beyond our sticky's and our TRELLO boards and things, what's out there that we could use to improve the accessibility of that information, help organize that information better, and also watch what's happening to those action items, the activities that we've defined. The features and user and functionality with defining that. What happens to that in the process of going from an idea through to actually implementation and functioning released production code. Let's say, production applications.

Damian (00:26:57):

So, there are a bunch of backlog management tools out there. Everyone's got their pet favorite. I am keen to hear, those on the call, what other people's pet favorites are. I think people who know me know mine, the tools I like and the tools I don't like so much. It is a personal choice. So, there's my disclaimer. We're going to continue with... In terms of perspective, is, really, what I've used over the 7 to 10 years, or tools I've used, what I like what I don't like, and why, more importantly. I'm happy to stand corrected if somebody wants to say, "Well, no, I think you're right about that." I'm happy to hear that.

Damian (00:27:47):

But this, I thought, in order to handle this subject, effectively, let's actually look at what characteristics, what kind of capabilities and functionality that we want out of our backlog management tools. All those sticky's and all those TRELLO cards, how are we going to manage those, keep a backlog, pick the ones that we want to work on that the organization wants to work on first, prioritization, watch it go through the the stream, the process of production, development, testing, QA... All the CICD type DevOps type of things we want and then ultimately get delivered out to the market. How are we going to manage all of that, with all the people involved, the development teams, the product owners, the product management side of the teams, organization, those kind of things?

Damian (00:28:40):

My number one pick for this... The first thing I'm going to look for is hierarchical breakdowns. I want to be able to break those high-level features that we came upon during our Agile inception or planning meeting. These conceptual things, features, [inaudible 00:29:02]. How am I'm going to better break those down in small chunks? If I can't do that, then I'm not going to be able to really visualize what the details are eventually. Whenever we set out on planning for a new product or a new set of features, we don't know the details yet. We're in that end of that cone of uncertainty, where we can have a pretty good idea. We might have some good developers on the team helping us estimate how big this thing is, but we're still in quite an unknown... The territory of the unknown.

Damian (00:29:35):

So, as we work through that, we want a nice easy way of breaking down that backlog but still relating it hierarchically back up to that high-level feature. So that, for me, is number one. I want a nice easy way of breaking down the ideas, the features, into smaller categories, smaller subsets, so we can deal with that and become more accurate with what we're trying to estimate in terms of effort and size, and all that kind of thing. So, that's number one for me. Number two, good visualization, I want a good UI. I do not want to click through 50 buttons to do a simple task, like break down my feature, add a set of tasks. I want to do things in line. This is true of any tool, it should not get in the way of the process, okay? We're trying to deliver value. We're trying to deliver new features and functionality to the organization so we can be a successful organization.

Damian (00:30:35):

Why would we employ tools that are a pain to use, they slow us down, everyone's getting confused and frustrated. We don't need that. The tool is there. Here's another definition of an Agile tool. The tool is there to support our work, right? To get the job done. We don't want to live in the world of the tool. Okay, the tools there, it's a means to an end. So, I want a really nice flexible UI. Now, I don't think there's a perfect one out there, but there's some good ones and there's some not so good ones.

Damian (00:31:10):

Metrics, of course, is a given. The whole point of Agile is to be transparent. If you can't surface important metrics around how the team's doing productivity and start getting predictability out of the team, then we could all go back to Waterfall or some other... Or, maybe Rational Rose, and go and do that. So, surfacing those metrics is huge. To me, when I first started working in Agile, and I was introducing these ideas, when Agile wasn't really well known... When I started talking to management about metrics, and being able to start getting predictable about how the team was doing on a two-week, three-week basis, and surface some information, they were just thrilled. They were thrilled that they had that visibility. There was a time when you didn't have that. You didn't know anything between the beginning of the year and three quarters of the way through the year. It was, hello, with pop ups nine months later, "Ta da, here's what you wanted, isn't it?" With the metrics, we can surface that on a regular basis, we can keep a track of things. So, super important.

Damian (00:32:23):

The next one integrations. Some backlog management tools live in an ecosystem and they can integrate really nicely with other tools, especially on the development side. Source controls, an example. On the DevOps, the continuous integration, continuous delivery chain. Those are important to us. Some tools do that, well, some tools have it baked in to some degree. The same goes for planning. Some tools have that now built in. They're looking at the big picture. So, again, some tools have that baked in, some you need to add on or some work independently. But, those two things come in four and five in terms of priority for me when I'm picking.

Damian (00:33:11):

Then six, characteristics six is a bit misleading, because I put it there, but it covers the whole thing. But, it does depend on where you intend to go. I've had organizations ask me, "Should we pick a tool that we can scale out? We know we can grow with it." I say, "Certainly, but let's start simple." When we're picking our tools, keep the Agile mindset right at the forefront, because Agile talks about simple, don't over build, don't over architect. Same with our tools, we'll deal with that when we have to. But, if we inherit a tool that takes monster configure correctly, for whatever you want to do, then that's going to slow us down. Tackle it a step at a time, a stage at a time.

Damian (00:34:04):

Clearly, if you anticipate that you're going to start it with two or three Agile development teams, and grow that out, start simply, and then get more sophisticated as you grow. That might require changing a tool. You have to deal with that when you have to. But again, try to think simply to begin with. Of course, if you want to scale out the box that will start with two teams, then we'll grow and grow and grow, then maybe you want to start out considering a larger scale tool to support that around your backlog management choices.

Damian (00:34:42):

Okay. I could pause for questions here, but I'm going to carry on because I wanted to just go into some examples of these backlog management tools and highlight some of the characteristics. So, this first one is Rally, Broadcom's Rally. Straightaway, I just wanted to show here's a breakdown of user stories. With Rally, you can break down a user story into sub-user stories, sub sub-user stories. You could start with an epic a feature. But, you can see you can do all of this in line as well, which is really nice. So, if you're in a planning exercise, an inception, you can have somebody typing away capturing this. And, because you can do this in line, you don't need multiple clicks to get there. So, I find that really helpful.

Damian (00:35:36):

As I mentioned, just earlier, the fact that I have an infinite capability of breaking that high-level story into smaller chunks when I need to, when I understand it better, it's there for me, I don't have to reconfigure anything. So, that's an important characteristic for me, is to look for that hierarchical capability without too many constraints. The next one is, again, this is a screenshot from Rally. Not that biased towards Rally by any means, but here's some tracking. So, a classic example, all our tools pretty much provide... That you can pick, pretty much provides something very similar to this view here, where we're looking at progress of user stories and bugs or defects through a particular time box interval through a sprint, through an iteration and that kind of thing. So super good, we can surface this information on demand, very useful in our daily stand ups. So tracking, super useful for the whole team, for the whole organization.

Damian (00:36:45):

This screenshot is from version one, which is another pet favorite of mine. A lot of similarities to Rally. I just like its simplicity. Here's another classic example of surfacing some metrics. So, we're all familiar with Veracity to understand, help improve our predictability for a team in assessing how long something's going to take. That big challenge with software. So, again, nice dashboards, surfacing those very important metrics for transparency, useful for the team, useful for the scrum master, useful for the product, product owner and useful for the organization overall, as you start aggregating this information together on the largest scale solutions.

Damian (00:37:31):

Then, here's a shot from Jira. Another classic shot. We can view our work activities in different ways. We can have the stacked version, or we can look at cards and things like that, mapping progress with cards. Jira does a really nice job of displaying that. The board, displaying the information and also customizing to really on the front of that card, to display the kinds of parameters and properties you might want to surface for the team, especially when you're meeting for stand up and things like that. So, you can grab a lot of information all at once, off the top of the card there. So, that's some examples of our tools in use across the board. I don't have time to go into every single nuance of it or other alternatives.

Damian (00:38:28):

But, to summarize in terms of backlog management, here are my top picks, okay? No surprises, I don't think, or maybe. Rally sits at number one. Why did I pick rally? I picked Rally for ease of use, visualizations. The simple side of things. It's not perfect. None of these tools are perfect. But, if I need to get a team up and running quickly, I always dive into Rally. I don't have to configure it too much. I don't have to customize. It pretty much works out the box. There's a good reason for that as well. That is, Rally was built from the ground up for Agile. It wasn't modified. He wasn't a tool. Well, it's not entirely true. But, pretty much, it was built from the ground up to support the Agile space. The Agile need to manage activities and functions and things like that.

Damian (00:39:35):

I'll contrast that with Jira and say, Jira does a good job. In every iteration of the release, it gets better and better. When I first worked with Jira, I found I struggled with it. Jira started out as a trouble ticket support tool for technical support. So, you create a trouble ticket, a technical issue, whatever, and you watch it go through the system. Then, when that thing was addressed, you could mark it as done. So, I know that the teams at Atlassian said, "Hey, we could use this for Agile." It's not a million miles away. So, they did. And, they did a really good job of marketing. Because, Atlassian has grown this fantastic ecosystem of tools for our development teams, development tools, testing CICD, DevOps, knowledge sharing with Confluence and things like that, they really have... In terms of market leaders, they're right up there. So, I'm happy to use Jira. I just find it quite a bit clunky. It's getting better, but it is a bit clunky to get things done.

Damian (00:40:47):

The other challenge with it is, you can customize the... I'm not going to say... I won't say that word, but you can customize there's something out of it, right? You can really create some fascinating, streamlined functionality, custom functionality for your particular organization. But, at the same time, you can get yourself in a right old mess, because nobody eventually... If everyone's going around and customizing it, nobody really understands how it works anymore. I've been on a couple of recent engagements with a couple of clients where I actually had to come in and rescue them, because they couldn't understand what was going on in their Jira instance. I spent the first two weeks saying, "All right, we're going to back out of this, we're going to just start from the beginning, get rid of all the customizations." Because, I think they had five or six people try and customize it over the course of a couple of years. It was scary.

Damian (00:41:43):

So, it's powerful. It's really powerful, but you've got to be careful what you do with it. You got to manage that carefully. I do like the fact that... I do and I don't like the fact that they've now integrating Agile Craft into it. I love Agile Craft. There's some interesting stuff going on in there and we'll see... That's a new for them. Atlassian do like acquisitions. They got TRELLO. I'm fingers crossed that they don't mess with TRELLO too much. I think, so far, so good, right? It's still a simple to use tool. Then, towards the bottom, Azure DevOps. I think Microsoft is really coming through with this now. It used to be TFS, but I've been using Azure DevOps more often in the last 12 months, and I like what I see there. So, that's a good player for me. It continues to improve in terms of capability and ease of use.

Damian (00:42:37):

Then, Version One, I have a nice empathy for, which is now digital AI, but they've always been there for me. Again, I matched them with Rally, but Rally's been my pet fave. Then, there's these other guys that... The Pivotal Tracker, Basecamp, Zepal, and things like that. Less known, but fairly simple tools. Quick go-to tools. So, I wanted to mention there-

Katie (00:43:01):

We actually have a question.

Damian (00:43:04):

Yeah.

Katie (00:43:05):

So, how would you go about switching if, say, you're already using Jira, but based on your recommendation, we want to switch to Rally? What's the process for that?

Damian (00:43:19):

Yeah. It's a good question. It depends how much backlog... How many things you've got, right, in your backlog in the Jira database? Let's call it that. It's possible. It's not easy, I'll say that. It's like, when people ask me, "We want to do Agile, is it, get this done in a couple of days, train these people up, and we're all good? I said, "No, it's going to... Yeah, you can do that, but it's not going to be effective. This is going to be tough, it's going to be hard." Let's say it as it is. Adopting Agile is not straightforward, if you really want to benefit from all the things that it can bring to you.

Damian (00:44:01):

The same with moving tools, it's not easy, but it can be done. There are ways of exporting the data, there's tools you can export to Excel, and then you can import into your other tool of choice. You've got to manage it carefully. It's something that you're not going to be doing overnight. You've got to plan for it and work carefully through that process. So, it can be done, it's not easy. But, if you're really struggling with a tool or that tool's not working out for you, do not continue to struggle with it. Okay? Because, it's just burning hours on everyone's time and frustrating all the scrum masters and the product owners. So, if there is a drive to say, can we look at something different? It's definitely worth going through that painful bit when you transfer. Hopefully that answers it.

Katie (00:44:57):

Yeah, I think it definitely did. That was a great answer.

Damian (00:45:00):

Cool.

Katie (00:45:01):

And, we're at 38 after, just to give you a time.

Damian (00:45:05):

Okay, so perfect. Let's move into the last section for today, which is the planning tools. As I mentioned earlier, a lot of planning is supported in a lot of those tools we've already looked, at one level or another. There are some other planning tools out there that are specific to the planning activity. How do we define planning? For those not familiar with that term, let's just quickly review what planning is. It's really that high-level activity. So, if you're beginning a new project, or you're developing a new product, or product line, or a new feature set, you want to start out with some high-level goals or high-level vision. You're going to break that down. Maybe, you're going to do some Lean UX on that, do some testing, test your hypotheses out, so you're not working in a vacuum. You now that you're building something that is really needed. I include that all in this planning exercise.

Damian (00:46:10):

If you're doing scaled Agile in any form, like Safe, everyone would be familiar, especially my St. Louis friends out there will be familiar with Safe and the big room planning exercises we did, with multiple teams, with multiple sets of product. It's a big exercise. You've got to manage that, you've got to capture all that information that's going on, so that ultimately you can go and execute on those ideas and those features. So, that's what I look for in a planning tool here is, again, we're going to break down those characteristics of what I look for in a planning tool. You'll see a lot of similarities here between what I look for in a product management, a regular backlog management tool, and a planning tool. Very similar characterizations here.

Damian (00:46:57):

Hierarchical breakdowns again. Now, with a planning tool, we may be even going higher up, we may be looking at initiatives, and portfolios and things like that. So again, that might be our highest level, and then we want to break those down all the way to, probably, user stories. But, we've got to look for that nice, easy way of breaking down your feature sets, your ideas, your vision down into usable, manageable, workable stories that a development team can execute on. The other things I look for on the insider, as part of a planning tool will be, the what ifs. Is there a capability for it for you, in a planning part of the organization, in the product part of the organization... Can you go and do a bunch of what ifs? If we do this, what happened there? What's the effect of that? So, I look for those kind of things. Those, what if tooling capabilities, to just look at different impacts of different approaches to organizing the backlog and executing on a particular release plan or feature set.

Damian (00:48:04):

The other thing I look for is dependencies, especially when you're scale, you're looking at what feature might be dependent on another feature, what development team is going to be dependent on that feature being ready, those kind of things. That's huge. So, if you have anything within that planning tool that allows you to map dependencies across different functionalities, different parts of the organization or different development teams, that's really helpful. Then metrics, of course, again, surfacing dashboards with progress, of the high-level of how those plans are being executed. Integrations, again, same kind of thing as our product backlog tools. What other tools does your planning tool integrate with? It might be a standalone tool, or it might have integration points to our development tools. Those kinds of things, to improve its capability and usefulness across the organization. Sorry, again, integrations.

Damian (00:49:06):

I think the last two, I think I've already covered. Visualization, again, I like nice, pretty pictures. Product Management, they want to get statistics, progress bar charts, and all that kind of, fun stuff. So, that's super important. Visualizing what's happening with your plan with the progress of executing on a plan, super important. And, enterprise and scale again, right across the board. If you're planning to really attack this on a large scale with multiple teams, one of the development teams, what support is there in your planning tool for that kind of environment?

Damian (00:49:49):

Then, a couple of examples of that in action. Some screenshots here. So, as I mentioned, some of our product management product, backlog tools that I've already mentioned, like Rally and Jira, and a few others, have planning built into them. They didn't used to, but they've got better and better at it, and here is a screenshot of the portfolio management capability in Rally. It's nice because it's all built into one tool, so your product side of the organization can view it this way from a high-level view and do some, what if planning. But, it's integrated into the development side of things, which is your user stories. Your Sprints, your iterations, you don't have that boundary, it's seamless that way. So, that's effective. And, they have some nice ways of presenting it.

Damian (00:50:43):

If you want it to break out and go full on with a dedicated planning tool, Aha! is a good choice for that. I would caution that it's a very sophisticated tool. It's brilliant at what it does, but you've got to accept the fact that it can get overwhelming. So, take it on with caution. But, if you really are, enterprise level, Aha! is great for that. But, it does take some configuration, it can do a lot of good things. You want to manage that carefully and consider how you're going to set that up. We've used that a few times, integrating Aha! It is standalone, although it does have things that do manage your day-to-day progress on user stories and things. But, we've tended to use it for that planning exercise and then integrate it to tools like Jira, or Rally, or Version One or something like that.

Damian (00:51:45):

Here's another example of some dashboards. So, Jira has its portfolio management and planning tool as well built-in. Very similar to Rally. That's pretty effective. They started out as add-on's, as plugins, to Jira, to do this high-level planning, this big vision stuff. And, it's got better over time. But, those are a few of the things that we'd look for. Few examples in the tooling area around planning that I look out for characteristically-wise.

Damian (00:52:15):

Then, here are my top picks. I'm predictable, aren't I, terms of what I like the best, but, again, my pet faves. I like the portfolio... They call it the same. Broadcom, Rally and Atlassian, call it whatever. I like the portfolio part of Rally, and I've used their version of portfolio with the Jira tools. There's some other things as well in Jira that support this high-level planning visioning. Then, there's Aha! Then, some other players in this mix here. Monday.com, they've done a great marketing job, as your go-to tool. I haven't used monday.com very much. A little bit here and there. There is a snobbery factor about using monday.com. "Well, you can't really be serious about Agile, if you use monday.com. That's like fluff." It's actually better than that. But anyway, there is a little bit of snobbery that comes into play there.

Damian (00:53:17):

But, those are my planning choices, if I'm going to look at planning tools. If I've made my investments in Rally, if I've made my investment in Jira, in Atlassian's ecosystem, then those would be my choices. Aha! for that kind of, standalone, high-level enterprise level planning tool.

Katie (00:53:38):

We have a question.

Damian (00:53:38):

Hmm.

Katie (00:53:42):

So, they said, "We spend a great deal of time planning in Aha!, only to change it all weeks later. What's your recommendation there?"

Damian (00:53:56):

Scream. No, well, we embrace change, right? What can I say? Agile, expect change to happen. No, I get it. We can all spend... This is what Agile tried to set out from at the beginning was, don't, for goodness sake, spend hours and hours and days and days planning. We've got to trim this down. We've got to slim it down so we're delivering value. Now admittedly, we want to be delivering the right thing. Okay, so we don't want to be working in a vacuum. But, there is a cut-off point where you say, you know what, we've worked this through countless times, let's just do it. You could really work your plans through almost too long and get too obsessed with the plan itself. The plan's going to change, it was inevitable.

Damian (00:54:56):

So, my advice here was just find a reasonable cutoff point where you say we've got enough here, the best way to move forward now is, to go and learn more about it, reduce those on [inaudible 00:55:06] and go and exercise on it. That's why I would deploy all the Lean UX protocols, practices, in this kind of case where, hey, let's get doing something, all right, and then we'll see. So, try not to procrastinate too much in your planning exercises. Just remember what Agile's all about? Again, don't let the tool take over, I think is the message there.

Katie (00:55:37):

Awesome. That was a great answer.

Damian (00:55:40):

Cool. Thanks. So, we're coming towards the end of the session here. Are there any other questions that I can help answer? Concerns, outraged people, seeing my bias towards Rally? Like I said, no tool is perfect. Everyone's going to have their pet faves. I would love to know if anyone out there shares my opinion or contests it, or says, actually, this is kind of interesting. It would be interesting to do a survey actually at some point. But yeah, anything out there?

Katie (00:56:19):

Yeah, there's two questions, actually. The first is about the cost of tools. They asked, "One of my clients just moved from one tool to another just to save money. Is there any recommendations on sticking with one tool?"

Damian (00:56:38):

Yeah. This is a really good question and it comes up a lot. I'll take some examples. Rally is not the cheapest tool out there for backlog management and planning, and Jira tends to be cheaper than that. So, you'll see a lot of migration. 9 times out of 10, if people are migrating... At the moment, I see migrating to JIRA mostly because of cost. But then, you've got to think about what are you saving. You're saving that monthly subscription fee, that licensee, whatever. In my due diligence exercises of looking at the reasons why we're moving, you've got to look at the usability of the tool. So, going back to the characteristics that I look for in a tool, by moving to this other tool, what am I going to lose out on? If I'm not losing out on anything, then it's worth it, and I save some money, if I'm going to achieve the tool.

Damian (00:57:40):

But however, if I introduce new logistical challenges, like ease of use, or button clicks... and, that sounds like so trivial. But, when you're doing hundreds of user stories, creating 1000s of tasks, and you've got all your scrum masters and your product owners trying to do this, you multiply the amount of time they're doing, getting frustrated and outraged and running off screaming... I've seen it, I'm not kidding. I think a lot of people on this call have experienced that, too. If that happens, it's going to be detrimental, okay? Cost-saving again, it may not actually transpire that way. It may not look it on the books, but if you look at the amount of time you're spent doing what you were doing before, but it takes five steps instead of two steps, you multiply the amount of time that takes. So, it does require some due diligence. Hopefully, using that set of characteristics I identified earlier for those tools, might help make that decision when you're assessing the alternatives. But yeah, cost comes in. Yeah, Rally's not cheap.

Katie (00:58:52):

All right, we have one last question. We tried integrating Aha! with Jira, but kept running into problems. Is there a better way?

Damian (00:59:05):

I know that problem. Is there a better way? Drop one of the tools. No, drop Aha!. Yeah, I've seen that problem. There's a couple of ways of answering this. One is, first of all, know exactly what you need to do on that integration point, okay? I can speak from experience that when we did this the first time, we weren't quite sure what we needed to do. So, there was some inexperience going in, and some guesswork. It was okay when one or two teams were in the mix there trying to work with the Aha! tool and the Jira tool, but as we scaled it out, problems started multiplying. Then, because we hadn't got our integration figured out properly at the time, we were so consumed by other things. But, we kept scaling and scaling with this problem still there. It's like technical debt, right? We hadn't addressed that, so got into a problem there.

Damian (01:00:22):

Then, we had to try and stop, hold everything, while we fixed those integration points, and then get better at it. Now, sometimes you're limited by what you can do on those integrations. But, allocate time to work through that, understand it well. Get some experience and pull in some expertise of people that have done those integrations before. Get that experience, get that training to do it, because it's going to pay off down the road when you use that tool more forcefully, at scale later on. But, it's always going to be a challenge at some level.

Katie (01:01:00):

That definitely makes sense. It's funny how those questions went together about the cost, and we're trying to make these two tools work. It's not just the financial cost, it's the emotional cost, too.

Damian (01:01:16):

Yeah, we're dealing with human beings. We've got our code and everything, but there's humans in the mix here, so we got to think about that.

Katie (01:01:26):

All right. Well, we don't have any more questions. I'll do one last call for questions. But, other than that, if there are no other questions, we will wrap it up here. We would love for you all to join us for part two in January, and Damian's going to give you a little sneak peek.

Damian (01:01:54):

Right. So, as I mentioned, part two, we're going to start... I'm actually excited about part two. Part one is your go-to stuff. Everyone's pretty familiar with those kind of tools. I think, hopefully, there's some takeaway from the session we've done today, that people might not have thought through in the past or experienced. So hopefully, there's some takeaway there. But, those are everyday tools. Part Two, we're going to tackle some slightly more interesting kinds of sides of what we do under the Agile platform. Distributed teams is obviously huge, especially now. What kind of tools do we have out there to improve communication between remote teams, virtual office spaces?

Damian (01:02:47):

Over the last six months... Just give you a quick example, I've tried to do Agile inceptions, right, which requires that real hands-on as you saw in that photograph, that picture earlier on, where you're all around a desk, and you've got sticky's... You're going around the room, and you're putting ideas together or breaking down user stories or features. That's such a human interactive and engaging... You need that engagement. How do you reproduce that virtually, remotely, and still keep people engaged? It's really difficult. Everyone's sick of Zoom meetings, apart from this one, of course.

Katie (01:03:26):

He says while he's in a Zoom meeting.

Damian (01:03:29):

Yeah, this has been a highlight of the week, right, the Zoom meeting, I say. But, how do we how do we engage people again? So, we're going to look at that in part two. There are some really nice tools, and I've used these tools recently on some of those very tight planning exercises, where having people around the room is so important. So, how do we replicate that as best we can? So, we're going to look at some of those tools that I found very effective.

Damian (01:03:56):

Knowledge sharing as well. Similar kind of thing with distributed teams. What are some of the tools that we can use to help distribute knowledge? Not just our wikis and things like that, but other tools out there that encourage team engagement. Then value, measuring the value. Anyone who's done DevOps and CICD and gone into that, we know what we're trying to measure there in terms of regular delivery, and the integrity of what we produce, the quality of what we produce. But, there's some other levels in there. There's some other things in there, nuances that we can start measuring as well, especially some of the new tools.

Damian (01:04:34):

I'm going to highlight a tool that we've got involved with recently, Tasktop, which looks at flow and flow frameworks and things like that. This is a result of all the work that was done... If anyone's read, Project to Product, that book is a superb book. I recommend that. But, the work that was done there has manifest itself in the flow framework at Tasktop. They do a great job of injecting things into our existing systems, measuring what's going on, looking for bottlenecks, and helping us measure value and where things are falling over. So, it's like a sophisticated CICD measurement, metric tool. So, we're going to look at that a bit as well in the next session. So, I'm excited for that, because it just takes us to that next level in terms of Agile tools.

Katie (01:05:26):

Awesome. Well, I'm definitely looking forward to part two, and I hope that everyone else is well. I have the link to register for part two in the chat and I also shared the link to that Project to Product ebook that Damian mentioned. So yeah, thank you, everyone, for your time, and we will hopefully see you in part two.

Damian (01:05:50):

Thanks very much, everyone.

Katie (01:05:53):

Bye.