



Digital Accessibility's Gap: AI to Bridge Mobile and Web Barriers

Michael Bervell & Jason Tan



SPONSORED BY

AKQA deque evinced  EQUAL ENTRY  Fable  aira

LIVESTREAM BY

 Accessibility SIG
Internet Society New York Chapter

Digital Accessibility AI and Mobile Barriers - Michael Bervell & Jason Tan

Accessibility NYC Meetup - July 8, 2025

Jason Tan:

Great to meet everyone. I'm Jason. I'm the co-founder and CTO of TestParty. Today we're going to be talking about mobile accessibility, which I think is a very underserved niche of accessibility, and with tons of users, but putting aside accessibility, I know that there's already fewer native mobile developers and cross platform developers, so I think it's just a space where not everyone kind of knows what's happening. And specifically, in accessibility, there are quite a few unique challenges.

I'd like to start with an analogy. I'm a big Classics nerd. I realized that actually I think the new Odyssey preview trailer just came out. I haven't seen it yet. But this was made a couple months ago. So I think of mobile accessibility kind of like the Odyssey, as compared to the Iliad. Where the Iliad in this case is web accessibility. It's a kind of much more well told story. People know the heroes much better. I think the Odyssey is an equally crazy crucible, and with a lot of interesting characters along the way. So... Yeah. I think of the Iliad as the web accessibility

journey, where you have tons of tools, you have tons of people talking about it. Even the AI discourse around accessibility is primarily web-focused. I would say.

In Odysseus's journey, returning from the Trojan War, for people who don't know the story super well, he encounters a bunch of different mythological creatures, and has to overcome these difficulties, in order to return to his home. One of them is the Lotus Eaters. Which is basically these group of people on an island, who offer up lotus seeds to his sailors and his crew, and happily, they're kind of like... I think they were very... Either starving, or they were just looking for the happiness that eating these seeds would provide. And instead, they, like, stayed on the island for way too long. And so I think that I jokingly put the Apple and Android logo in there, because these platforms make a lot of promises. Right? In terms of their mantra -- it's kind of a lock in first kind of system. Versus the web's open nature and kind of the standards that were developed around it. It's a little similar to the Lotus Eaters.

And Polyphemus, which was one of the giants that kind of troubled Odysseus, he was in a very dark cave and had to try to navigate while this giant was swinging his club at him. And I think that developers in general just feel a bit lost. Like Odysseus was in that cave. And a lot of factors contribute to this, specifically for mobile. And we're going to go into some of the slightly more technical parts of that.

And then finally, we come to Ithaca, hopefully -- today, we're trying to present not just me, venting about the problems of mobile accessibility. But hopefully offer a path forward, so that we can achieve the promised land here.

A little bit about who I am. I studied economics and then computer science and Latin. Hence the mythology and Classics nerd. I was an engineer at Twitch on their iOS team. Helped them kind of fix a lot of their code on the mobile side when they got sued for accessibility. So a lot of firsthand experience on this topic specifically. And then now I'm the founder and CTO at TestParty, where we're trying to use AI and non-AI techniques to build better automation around accessibility testing and remediation.

So a little bit of the contents here. We're going to skip through some of the slides that are very company-specific. Today we're going to keep it pretty theoretical. But first we're going to talk about the present. Then we're going to talk about a little bit of our solution, and then how it addresses a lot of the advice. I think we'll probably just pause after we go through the solution and take questions from there, since I think today is only a 10-minute segment.

So the present -- what I want to talk about are really... First I like to use a survey just to kind of get people primed on the mobile accessibility statistics. Does anyone know this? Off the top of their head? Of what percent of iOS users use larger text? I wish! I wish! But... It's not that high. The actual answer is around 20%. So I think... It's much higher than I would say most accessibility statistics that you've memorized for any sort of CPACC exam -- it's much lower than the general population. But it certainly is a statistic that is worth paying attention to. I think one in five is humongous already. And typically, when I look over the shoulder of someone on the subway, I can usually find someone with enlarged text. So... This was what I could find online. It might have inflated these days with increased screen usage and deterioration of retinas. But 20% is the quoted statistic.

There's also a survey published by the American Federation for the Blind, specifically around mobile accessibility. Now, there's no real trends that I want to point out here. Certain sectors are more conscious of mobile accessibility than others. This survey did a great job of kind of looking at very diverse application groups from crypto to banking to ordering. But all in all, you can see that the accessibility is still quite a large barrier. Whether -- here "occasional" is defined as less than half the time, whereas "frequent" is defined as more than half the time. So for the majority of apps, you're still seeing around 60% to 80%+ encountering an issue -- if you add up the occasional and frequent. So you're seeing 60% to 80%+ of apps having issues on the mobile accessibility front.

So I categorize the four main issues of mobile accessibility into the four apocalyptic horsemen. The first one is around standards. Right? So a lot of the room might be familiar with WCAG, Web Content Accessibility Guidelines. I cheekily ask: Where is MCAG? Mainly because in order to have a really thriving ecosystem of developers and Open Source tools, you need to have, like, a unified standard. I know that the W3C has released a working paper, working draft, on mobile accessibility, which adds a load of different kind of controls that otherwise wouldn't exist on web. And I think that that's really important. So if we go to the next slide... Yeah. I think just this principle of vague rules creating vague solutions -- there's a couple companies that I want to call out, that are just doing really great work, in terms of creating standards. Like CVS Health, T Mobile. I know that a couple larger companies have released mobile accessibility guidelines that are specific to native frameworks. Because things like Swift, SwiftUI, Kotlin, they just simply don't have the same behavior as, like, adding ARIA or something to existing HTML or React or whatever. Most of those APIs are, like, very specific. And if you think about the controls, whether it's swipe, long press, double press, things that just don't exist as very

popular UI interactions, that otherwise the web can kind of get away with, with just simple buttons and inputs.

The second thing that makes this really difficult is lack of automation. So next slide - I show you some of the most popular web frameworks right now for automation. Many of you guys have probably used it in end to end testing, or if you're at an AI startup or something, this is the backbone, really, of AI agents navigating the web right now. And on the next slide, I show you what we have for mobile. Which is Appium. I don't know if there are any mobile developers in here. This is like the Open Source equivalent of Puppeteer or Playwright. It attempts to connect to emulators. You can write end to end tests for it. It's very difficult to use. It's got 21k stars, but it's an extremely difficult to maintain framework. And I would say that many companies choose to even not have UI tests, simply because this is very, very flaky and very hard to have the right amount of engineers who have expertise to write for this framework.

I think the next slide I talk about, really... Many people are following OpenAI and their announcements of a lot of features. What you haven't seen from OpenAI is any sort of attempt at mobile automation yet. Because -- and that's just -- you know, me reinforcing the thesis of... Without kind of the right Open Source automation tools to begin with, it's very difficult to start systematically testing. On OpenAI, with Operator, you can basically tell it to try to book an Airbnb, and what it's doing is it's using Puppeteer and Playwright and screenshotting things and navigating and you can't do that on mobile unless you're very crafty and trying to overengineer a lot of things.

And the third point I'll jump to is what I'll label as opaque element structure. So a lot of people... I'll shout out axe-core from Deque -- is really one of the best Open Source frameworks out there that primarily relies on embedding itself within the HTML DOM. And there is no such DOM in mobile. If you've tried maybe cross platform mobile, like React Native, you might have seen the React Native debugging tree. It is technically available for you to try to crawl, but that is nowhere near the accessibility provided by the HTML DOM. If you go to the next slide... DOMs are really crucial, I think, to constructing the... Like, the relationship between elements on a screen. Without this sort of representation, you pretty much have to rely on simple visual analysis. And computer vision is not quite there yet, in terms of reconstructing entire views from just a picture. And reassociating that with code. The code, fundamentally, beneath it, is where the information lies. And so... Yeah. I'm just showing for the people who I know I've met a couple people who are not engineers... For DOM, it's like the Document Object Model. It shows kind of the

structure of how a page is overlaid. You can see this element has padding. This one has margins. For mobile, I screenshotted, I think, xcode, where you basically have a view hierarchy. Right? And it's totally useful, in terms of debugging things that are just... You're functionally looking for things that are not showing up. But it is very not useful for an automated test, looking for... You know... Is something... What is it labeled as? Or... The accessibility attributes of something. You just simply -- Apple hides a lot of that. And you don't have access to it.

And that segues perfectly into kind of the platform limitations. Of... If we go to the next slide... We've noticed that Apple and Google have a certain level of restriction on their APIs for what you can and can't do. And so they have... You know... They have a lot of prebuilt components. They only have really a couple fields available for accessibility. Some of them are very, very useful. Some of them are... I think way too restrictive. And that also makes it really hard to navigate, like, how to make things accessible.

Oh, okay. That was way faster. So... I'll do a summary version of our solution, since I don't want to get too company-specific. We've now built on top of, like, a better automation framework that really kind of strips away the ugly parts of Appium to basically create a new standard that is much closer to Puppeteer, where you can kind of write a YAML-based test file, and navigate through screens, and create hierarchies around them.

So we have, like, so what's buried deep inside Apple and Android are actually -- you can capture the element types and the various relationships inside of a hierarchy, and what we've done is now we've created a way to use that hierarchy for real automated testing. And so that's really where this is going, is like -- how do you control the mobile emulator, run tests on each individual screen, and still allow, like, the simulation of, like, a user journey? So we -- yeah. Some of the screenshots that we have are from an extension that we're building, and it wraps around a lot of the core, like, UI inputs that a lot of other automation can do. But what we do that's very different is that we can do it while detached from the Apple ecosystem or the Google ecosystem. Like, this is strictly within VSCode. So we're basically taking full control of the running of a device, and you're able to test things remotely. You don't need to kind of blend it into an existing end to end test, which I alluded to is very hard to maintain on mobile in general. You can kind of out of the box simply create, like, user journeys, and then do accessibility tests on them. And they're catered towards mobile in the way that they're very element-specific right now, rather than... I think a lot of WCAG or aXe-core is very attribute-specific. Where in mobile right now, there aren't that many ARIA attributes to really validate against. You're

not looking at various like... ARIA checkstates, ARIA livestates, all of that. You can only really look at a level of sophistication above that, which is like... Input elements, date pickers, at the functional level. So... We kind of group that into actually -- through a lot of the hard work of CVS Health as well, they've kind of defined a lot of the standards, using the various components that are native to each platform.

So I'll pause it there. We want to just make it so that it's very automated, and it brings, like, true developer joy to accessibility. I think that that's primarily one of the biggest barriers, I think. Is that in order for mass adoption from developers, a dev tool needs to be extremely easy to use and extremely fast to automate. So I'm happy to take any questions there.

Ben Ogilvie:

We're going to break into two segments here. So Jason is going to take questions now. A little bit different than we usually do. And then Michael is going to give his talk, and then we'll do questions for Michael. So... Any questions in the room for Jason real quick? I'll look and see if we've got anything online. Oh, yes.

Audience Member: Going back to the 20%, does that also include users that blow up their phone using enhance? Because some might not know about the larger text feature?

Jason Tan: I don't think it's including that. I think it was... Yeah. I think it was specifically just -- and this is just iOS. I wasn't even counting other platforms. Other phones. Other platforms. Yeah.

Ben Ogilvie: Because that stat is from the Apt Foundation's SDK, 42 SDK. That one is measuring specifically having the enlarged font set at the OS level. Any other...

Audience Member: With the rise of things like PHP and React Native, how well does this tool integrate into those frameworks?

Jason Tan:

Yeah. That's a great question. Because I do think that cross platform is a very popular feature. Yeah. This new tool -- we didn't build the mobile automation framework. It's a tool called Maestro. But it's very cross platform. Not just with React Native, but also Flutter, Maui. I wouldn't recommend building Maui apps these days. But yes. It's quite compatible there too.

Audience Member: Great talk. Thank you so much. I was wondering... How do you balance automation with human feedback? In terms of accessibility pipelines?

Jason Tan:

Yeah. I mean... I think... For web, most people are quoting low... 30%, 25%, for what automation is covering right now. Deque covers I think... 50% is their stat? Mobile is very... Mobile... We're in the very early stages. I think there's zero doubt that there's a lot of human input here. I think just the mobile QA process in general is still very human driven. Especially when you're dealing with, like, tapping... The various controls that I talked about. Still quite hard to consistently simulate that. You've had startups that are trying to, like, do robot-based tapping on phones and stuff. And that still hasn't really become consistent and taking off yet. So... Yeah. I think it will stay at around, like, 80% manual, 20% automation in mobile for a while. I think web could take off beyond that much sooner.

Audience Member: Why do you think web progressed so much? Is there a reason?

Jason Tan:

I think it's the maturity of the tools. The maturity of the... And the openness of the framework. Right? So I think the HTML framework, all of the tools out there are basically Open Source. Apple is actually... I think they just released two months ago their Swift compiler so that you can actually build Swift apps outside of Xcode much easier. So we're like a decade behind on the types of tools that have been released for mobile.

Ben Ogilvie:

One from online. Does PreGame integrate with Android Studio?

Jason Tan: The IDE specifically? The Android Studio IDE? No. I think that's also a very interesting challenge of mobile, and that's part of what I complain about, I guess. Is that: With Apple and Google, they've chosen to lock you in to IDEs like Xcode and Android Studio. Even in those IDEs, you can't suggest code as easily as VSCode, which is Open Source. AI code gen tools are choosing to avoid Xcode and Android Studio specifically for that reason. So I think there's a day that we will try to integrate. But today is not that day.

Ben Ogilvie: Any other questions for Jason? Cool. All right. Switch gears to Michael.

Michael Bervell: We had Jason go first, because his is technical and mine is fun. Yes. We'll go to the next slide. No, Jason's was fun too. I think I always enjoy hearing

about mobile stuff. Let's go to the next one after this too. Today we're going to talk about automating the WCAG. If that's possible. What's possible? Jason was all about mobile. I'm talking all about web. And we do see them as two separate... Probably parts of the same coin. Right? One is the front side. One is the back side. No, there's nothing on the slide. Yeah. That's on purpose. But we're super excited to share a bit more about how we think about web, in addition to how we think about mobile, and it is very different and distinct. But obviously I always like to start with a story.

So I'll tell you about me when I was in fifth grade. One of my favorite TV shows was Teen Titans, with Robin and Raven and Star Fighter and Beast Boy and Cyborg. These were these teenage superheroes who would just fight like super villains. And of the five, I had one who was my favorite, and my favorite was Cyborg. He was this half cybernetic robot, half man, could shoot blasters out of his hands. That was just crazy. It was so cool. But he was more than just a cyborg. He was a Black, technology loving disabled superhero. And in reflecting on this talk, I realized this was one of the first and probably only times I had seen a disabled superhero on television. And this was a TV show from the '90s. What does it mean? What was Cyborg? That's the investigation of this talk.

Before he was Cyborg, he was Victor Stone, this athlete, a strong teenager, who his mother -- eventually passed away in an accident that left him injured and his father being a technologist decided to keep his son alive using cybernetics. The effect of that was that we had Cyborg. This half technology, half man assistive device assisted guy who could do more in the world with assistive technologies.

So that's kind of the theory of today's talk. Right? That Victor Stone plus technology plus his humanness made him this superhero who we called Cyborg, this superhero that I fell in love with and still really love. When I was in 6th grade. And I'll argue today at least one thing that we think about is that AI and the way that you can use AI is a way to turn yourself into a sort of accessibility superhero. That you plus these assistive tools, which are AI tools, plus your own heart will get you into becoming this sort of superhero. So in the next... Let's call it 14 minutes... I'll run through all three parts of that. I'll talk about the research behind it and talk about some suggestions on how you can use AI tools to be a better sort of superhero.

But first, about me. Who am I? I'm a nerd. This is me. Thank you, thank you. Yes. I'll take the applause. This is me and my brother, living out our real life superhero fantasies at Chuck E Cheese circa 2004 in Seattle where I grew up. I continued to be nerdy as I got older. I loved computer science. That's what I ended up studying in college. I loved reading. So I studied philosophy. I started donning glasses very early

in life. You know, I actually didn't have to wear glasses growing up. I think I was farsighted or something. And I just wore them all the time, to the point where my eyes are terrible. So that's... So now I wear glasses all the time. It's great. I love it.

But who am I now? I'm the founder of TestParty and CEO of TestParty, along with Jason. I studied computer science and philosophy at Harvard. Before that I did accessibility consulting at the UN and Google and worked with Microsoft, and we invested in companies like OpenAI and Kahoot and that was a fun journey. And yeah? I'm still a nerd. That's not going to change.

About this talk. With TestParty, I think of myself almost as Cyborg. And the reason for that is mainly because... I really truly believe, and what we hope to argue today is that by automating the WCAG, at least what parts of it you can, specifically remediation. Not just finding issues, but also fixing them, you can use AI to also become this digital accessibility superhero. So I'll break down all the parts of this thesis statement.

By talking about: Where are we today? What can AI actually do? And what are some practical examples of how AI can help remediate the WCAG? Then we'll talk about the impact and take questions. And you can all go home and take a good nap and reflect on this talk forever, hopefully.

So where are we today? This is chapter 1, episode 1, season 1. Yeah. It's the pilot. This is the pilot. Where are we today? Today... The internet is inaccessible.

And 95% of the million most visited home pages are inaccessible. I think this stat might now be like 94%, as of the most recent. So we've made some progress. Which is good. The web is also becoming more complex. So if you look at the number of elements on these home pages, this year has 11.8% more elements, meaning there's more 11% more things to test, than last year. And we anticipate that will continue. The web will get more complex, people will get new creative and do it with more interesting new components. And the idea is manual auditing. Not just accessibility but security. Pen testing. TestParty went through SOC 2 compliance. We were told it was going to take 8 weeks. We finished all of our parts in 8 weeks and it took another three months to get the rest of the audit, the rest of the report. This is common for accessibility I'm sure as well. As professionals, you have a lot on your plate. You have so many audits and you only have so many hours in the day and so much energy. Right?

So they take months to complete. And the effect of this is that there's all these sites. Specifically ecommerce websites. We've been diving a lot into ecommerce, working with ecommerce companies, and they're getting sued at super high rates, to the point where in the last five years alone, we estimate \$400 million have been paid in settlements and lawsuits. This is a whole other talk where we talk about what it means for a select few firms to be doing this litigation work and who is funding that litigation. Why is it funded in this way -- that's not this talk. This one is more fun. And of course if you look at people in this room and people on Zoom, I wouldn't be surprised if you admit you have burnout. And one of the studies that this came from is from Devon Pershing who in 2023 said 60% of accessibility professionals experience burnout. I tried to question: Why is this?

And if you think about it, we can just go back to Teen Titans. As it all leads back to. Cyborg used to argue -- when I was an athlete, human, I loved pushing my limits. Getting stronger, faster, and better, just by trying harder than I had before. And his coaches would always tell him to give 110%, which he did. I think we're in a very similar position. Where we're giving our 110%, we're experiencing burnout, but we're only getting out of it 5%. The 5% of the web that's more accessible than it was last year. Or maybe now let's call it 6%. Are we okay with that?

So my argument really is that we could do better than 5%. And one of the ways to do that, if we go to the next two slides, is to start using some sort of artificial intelligence, some sort of technology, just like how Victor Stone had his technology to in essence help us out. Right? So this leads us to chapter 2. Which starts with a poll. Of the room. How much of the Web Content Accessibility Guidelines do we think that AI can automatically remediate? Okay. Put your hands up if you think it's 5%. At least 5%. At least 5%? Yeah, yeah. Okay. Cool. Keep your hands up if you think it's at least 20%. AI can fix 20% of bugs, all that alt text and color contrast. Okay. 50%? Half the hands have gone down for people on Zoom. How many people think at least 75%? Okay. We have two dreamers in the room. How many think 100%? We can automate 100% of the Web Content Accessibility Guidelines? Hands have gone down. One day. There we go. Both. Both.

Yeah, yeah. Yeah. So this is a good point. If you go beyond the algos... Ooh! Predicting the talk! Okay. We'll go to the next slide here. So let's talk about what AI can do. Right? So we are all on the same page. When we think about AI, let's first talk about what it is. AI really is -- at least as we define it today -- if we push one more, there's a picture here -- I define AI, at least in the context of this talk, as these large language models. Things like... ChatGPT. Which has o1, o3mini, o4. Gemini. Google's sorts of tools and technology. And these large language models essentially

work, as we all probably know, by taking huge amounts of data, adding weights to that data, in terms of how they interpret it, and making predictions. And so these predictions are actually getting quite good. Right?

So if we go to the next slide, we'll be able to see that all of these models can score better than me on any standardized test. Which is kind of a funny realization. Why do I need a lawyer when my AI model has an LSAT? It got 80% on the LSAT. Why do I need a world historian when my AI model gets 100% on the AP World History exam? These benchmarks are showing that these AI tools are getting better and better and better at what I'll call general logic and general reasoning. I was leading a research study when ChatGPT first came out at the Harvard Business School. The study came out in September of 2023. And I'll walk you through some of the findings, because it does kind of delineate what types of learning AI is best at versus worst at.

In the experiment, we took 80% of BCG's global workforce. That's the Boston Consulting Group. And we had them do 18 tasks around creativity, analytics, and persuasion. We gave half of the group access to ChatGPT. Specifically GPT-4. And half of the group no access. They were just regular humans. Victor Stones, if you will. What we found was that for those who had access to AI with no training, they were able to work about 12.5% faster... They did 12.5% more work, 25% faster, and 4% higher quality. That's what this graphic is showing. The difference between people who didn't use AI versus those who did use AI. If you look at just those who used AI, there's also an interesting barrier. Some people were 10% better than their other AI-using counterparts. So as researchers, we were interested in... What was that difference? That was quite obvious. And there were one group of consultants who called themselves centaurs. We called them centaurs. They just called themselves people. They were mythological half horse, half human creatures who divided the work between themselves and AI. It's almost as if you were to use a calculator and say... What is 2+2. And it would tell you the answer is 4. And you as a human would use 4 to do the rest of your work. I would describe that as centaur work. The AI is an assistant to your own thinking. Versus the other half of consultants who performed better were what we called cyborgs. Where they completely integrated AI into their workflow. So they were continually interacting with technology. The best example of this that we found in the study, because we had all the chat prompts and how they were talking, was that cyborgs were prompting on average 3.5 times more than a centaur. So a centaur might say... Hey, I have to design this new marketing campaign for Nike. Which is one of the questions. Give me ten example names. And a centaur would just say... These are

the names. Whereas a cyborg would actually ask and say... I like name number 5 and 7. Give me more that are like that. Okay. Now based off of this, refine it to another subset. And that interaction, where you're using AI as almost a sounding board or a teammate, which was the next study that we did, those sorts of performers performed way better than just the ones who would ask AI questions, almost as Google search.

So the results were that we found, of course, not just with an AI, but just generally, that those who used AI did a lot better if they were low performers. So as a low performer, I'm happy about that. Because it means that AI will help me more. But we did find that people who were already skilled did still benefit. So people who were not experts had about a 43% increase in perceived output. This is just of the AI users. And of course, for the top half, it's about a 17% increase. So what does this mean? I want to highlight just how kind of drastic of a change this sort of number is, when we say 14% more productivity. Or 23% faster.

Historians estimate that when steam and power entered factories in the 19th Century, it improved performance by about 18% to 22%. And so now that we're seeing AI, which is doing mid-double digit improvements, that's significant. So I just want to highlight that. Just for context. To put it in... Not numbers, but in kind of historical relevance. So there's two takeaways from this that I want to share. First is that the expert AI... Quite honestly... The expert AI is going to be better than the average human.

And really expert human is still going to be better than the average AI. Right? Which is good for a lot of us... What I call T-shaped people. People who have a lot of different interests but one focus area, AI may beat us in a lot of other interests, but that focus area is where we're going to win. And second, how you use AI is much more important than whether you use AI. And I bring this back to this cyborg versus centaur. Are you using AIs as a frequent sounding board or a one and done, a one shot tool? And one of these ways, the cyborg way, is much better than the separated centaur way. These are the two major takeaways and I want to argue that this is the right way to automate the WCAG. I think I have two minutes before I need to open up into Q and A. So I'm going to speed through more of these slides. When we think of what the new experiment is, one of the new experiments is just trying to be more practical. Bringing this into our day-to-day life in a corporate setting.

There's a couple of examples of how you're probably using AI tools. Whether it's Grammarly, Be My Eyes, GitHub Co-pilot, Cephable. If you don't know these

startups, Google them. This is the frontier of what AI accessibility is going to look like. As it applies to accessibility... There's a good quote here. We can share these slides afterwards. But I do think there are a lot of questions about: Can you automate the WCAG? And we're talking not just about... Testing, but remediation. So I know I asked you this question earlier. And I decided to ask AI this same question.

I asked AI: How much of the WCAG can be automated with tools like large language models? So DeepSeek said 70%. 20% could be fully automated. 50% with human in the loop. Claude said 75%. So it was a bit more ambitious. The dreamers. Claude is a dreamer too. 17% fully automated. 30% human in the loop. ChatGPT said 80% automation. Thank you, Sam Altman. 50% automated, 30% human in the loop. Gemini said 95% automation. Actually, originally it said 100%. And I said come on. And it said... Okay. 95%. It's 95%. I think it was DeepSeek. 70%. Yeah. The most. Exactly. Yeah.

But what can't AI remediate, right? There's a lot to think about here. And again, we're talking about remediation, not testing. I think that's really much more exciting in my opinion. What can't AI remediate when it comes to Web Content Accessibility Guidelines? I'm lazy, so I asked AI what can't you do? And it told me this. It said... Hey, if you look at subjectivity versus automation feasibility, that's the metric that you can start to weigh to see what we can and can't automate. So things that are like... For instance... Super subjective. But very low automation... Is the hard part. That was the ChatGPT answer. I asked Claude the same question. And I had another matrix of, again, contextuality, versus automation potential. Which I feel is not fair. If I was like... How automatable is this and one of the metrics... Axes... Is that it's automatable... I don't know. Thanks, Claude. I'll move to next one. Gemini said if it's a basic pattern versus an advanced pattern, simple and rules-based versus more contextual and complex. So really if you summarize this, there's a number of things that these models are predicting that it can and can't remediate, and really the summary as I think of it... It's really that the experts in this area are always going to be valuable, because the AI can't replace the expertise. That subjective judgment, the... How automatable is this metric... Really is the intangibles. The idea of your heart. It's what you're doing when you're at your best. In the same way that Victor Stone with his technology still needed to have ethics and morals, or else he would have been a supervillain. We're in the same category here as accessibility superheroes. I know we're getting towards the end of the time. So let's go to one of the examples that we have in here. Keep going. Keep going.

Keep going. Actually, go back to three or four slides. Let's talk about human in the loop, and then I'll wrap up there.

One thing that I really love to think about is human in the loop. It was a question earlier. How do you moderate these AI systems to ensure that the outputs are actually accurate? And we argue that the best way to do that is through what's called human in the loop. And that's where the centaur versus the cyborg mentality really is the most different. A cyborg is constantly involving themselves in the loop of the AI interaction. Right? When I'm prompting 8 times in a row, I'm validating and reprompting and revalidating. And so there's something to be said here that having a human in the loop is quite important.

And we're pretty familiar with this, I think, anecdotally, already. As people. Right? Because we use human in the loop technologies all the time. Right? So if you use spellcheck, that's a human in the loop AI technology. If you use fraud alerts. Right? When you buy a \$7 smoothie from a weird store and Apple is like... Are you sure you want to buy this smoothie? Yes, I am. It's a Tuesday night. You know? Random example. And of course, autocomplete is another one of these types of human in the loop technologies. If we Zoom forward six slides, and this is the last thing I'll pitch before I open up to questions, I like to think a lot about AI agent in the loop. Right? We talk a lot about human in the loop. And if we're really going to the point of full automation at some point, how can we have an agent who is looking at results and validating results, versus just us as humans? And so this is one thing that we think about quite a bit at TestParty. How do we use technology to validate technology? In order to create better and more effective models within a specific niche?

This niche right now is accessibility. But you can imagine using that same process in any type of niche. Security. Design. Usability testing. And that's really the vision of where we think AI is going to go, as it relates to kind of these human-focused design aspects. Right? And so then really... The question then becomes: How good is your validation agent, right? And I could talk about this for hours. Right? There's examples of positive validation agents. Where you're saying: Yes, this looks right. And there's examples of negative validation agents. Where you're saying: No, this looks wrong. And if you have an agent who's trained on being a negative agent, they can give feedback in a different way that a positive agent can. And you put them together and you have a system that's more of a cyborg AI system than sort of a human in the loop centaur system. So that was a lot of mumbo-jumbo of what we're building. It's whatever. But... Let's open up to questions.

Audience Member:

It's not mumbo-jumbo. It's your personal thoughts and reflections. And I love how the two of you have this theme of mythology and literature. Because it's the one thing that AI doesn't take into consideration. There's a lot of ancient wisdom that we've overlooked. And history is going to repeat for those that don't learn it. So thank you for the reminder. Getting into the meat of the question... These models, the automation -- you asked them, but what were the parameters that they used to make the conclusion? Ultimately they were dependent on the knowledge graphs and the prompts. Chances are, if I asked the same question, I wouldn't get the same answers.

Michael Bervell: I was very generous. I said: Assume you know everything. Meaning you have full access to the codebase. You can test these tools. Like Jason described Maestro. You have access to Maestro and you made it work in the ways you assume it would work to be the most effective. I essentially assumed full access and full knowledge. Which you can argue is kind of what a human would have. So that was the parameter on that question.

Audience Member: Hi. So when you talk about integrating AI into the evaluation loop, are you just considering in English? Or like... Is it also applicable to other languages?

Michael Bervell: Yeah. I'll answer this question with a reflection. When ChatGPT first came out, it was trained on the global... All of Wikipedia, essentially. The whole internet. And no one really trained it to speak different languages. But you could chat in whatever language you wanted to, to ChatGPT. And it would respond back in that language. And so I assume -- at least in this case -- depending on the training dataset that you give to your model -- it will still be able to do the agent evaluation in whatever language. Right? Whether we're talking like... English, Spanish, French, or we're talking React, JavaScript, TypeScript, or the language of pixels. Or the language of pictures. Or the language of video captions. Right? You can start to redefine what a language is with a lot of these models. That's where a lot of the innovation is coming from today.

Audience Member:

Have you noticed if different models perform better with different success criteria from the WCAG? And also what levels? Like A, AA, AAA?

Michael Bervell:

We talked about --

Audience Member: Or perform worse.

Michael Bervell: We talked about this at our all hands today, actually. I think if you go forward four or five slides, it's in here somewhere. No. It's not. Never mind. I used to have a slide in this deck that was describing... Like... How good an image alt text was. And you can ask ten different models to give you an alt text for a picture. And they would all be different. And then you could weigh it, based off of: How good was the output. And there were better models at alt text than others. Is essentially the long story short. I'm sure that would also apply to these WCAG rules. There are some models that would be better at some rules than others. And even one of the first graphics I showed of performance on the SAT and LSAT and GRE, certain models perform better. Definitely -- because no one really knows what's in the training data and no one really knows the model weights perfectly, there's some level of unpredictability. What that seems to apply is that you could in theory fine tune these things could be really good at certain WCAG criteria. Which may or may not be what we're doing.

Audience Member: Thank you.

Ben Ogilvie: All right. I saw another one over here. Hey, Xian.

Xian: Hey. I miss you guys. So hi. I'm going to say that I'm not the technologist in the room. Which is funny. Because I'm chair of the New York Tech Alliance. But don't let that fool you. So I'm going to ask the most obvious question that I think most lay people wonder. Is... What does this mean for... What do you feel your findings mean for jobs and integration in terms of the human element? So this is the Captain Obvious question. But...

Michael Bervell:

I think the biggest takeaway is that expertise matters more now than ever. Right? Because let's suppose you really want to go the route of agent in the loop. For evaluation for everything. Right? Meaning that you have an AI agent that reads your website code and suggests fixes and you have another agent that tells you if those fixes are right or not. This is really... When you see people talking about MCP servers, that's kind of the route that that's going down. That can only be... Someone has to make the agent. Right? Who is going to do that? The expert. The expert in saying... This is right. This is wrong. Even if you're just training your apprentice AI agent to do the work for you. Right? So I think really what that tells me is that we're all in the right place at the right time by being experts at something or desiring to be an expert at something. And that that expertise is never going to really go away.

I guess the things... All the slides I skipped were the... What? Why does that matter - slides. So you asked a good question. Because I skipped all those slides. But really the key takeaway is there's a certain level of humanness to accessibility that a lot of other industries don't have. And so we need to retain that sense of human ability. That the work you do matters. Because compliance is just the floor. There's so much more that we can do, that we may not be able to train AI on.

Xian: Well, because I have friends who do this by hand, and so I'm wondering... Does this make them obsolete? Are they needed to make sure that it's doing what it's supposed to be doing? It would just be great to know kind of... Obviously you mention that we do need the experts. But do you think that it could make those jobs obsolete?

Michael Bervell: I think it'll probably change the jobs, for sure. I think the "this" that we're describing would be fundamentally different. And I wish I could predict what that looks like. But I don't know.

Ben Ogilvie: Yeah. And I'll kind of key off of that. We've got just a couple minutes left. I'm going to take a couple online and maybe one more in-room. And we can... Those staying around -- we can continue the conversation afterwards as well. But... One question online was kind of a healthy level of skepticism. LLMs have trouble with hallucinations. Can we really trust them to do accessibility remediation? Kind of keying off of that where does the expertise lie question.

Michael Bervell: I think certainly LLMs hallucinate. And I think the solution to that is -- you can call it an antiagent. What I'm talking about... Where you have an agent whose whole job it is to root out hallucinations. So you say: Assume you're an expert in the WCAG. You are looking for commonly hallucinated errors for WCAG. Look at the results of this other agent and tell me on a scale of 1 to 10 how accurate or not accurate it is. Right? And so... When you start to stack 100 of those types of agents on each other, to look at the same codebase, maybe specified to certain WCAG criteria, you know, you start to get a different level of accuracy. Versus... You just ask one agent and take the one result as true.

Ben Ogilvie:
I think that's a really important... The adversarial agent model.

Michael Bervell:
Exactly.

Ben Ogilvie:

The next question said: I find myself skeptical of percentages the LLMs provided, because we put some HTML and PDF through Co-pilot. It remediated 70% of the errors we introduced, but many of the remediations were wrong. So that's where the adversarial model... Claimed victory. Needed something else to refute it.

Michael Bervell:

I trust AI to write my emails. And that's about it. For now. And also rewrite all my code. But that's fine.

Ben Ogilvie:

Do you include people with disabilities for testing your product?

Michael Bervell: We do. We usually hire a specialist agency. To your question earlier, there's always going to be a need for specialists at some level in the test process. Accessibility is probably 10 years behind security. And even in security, like a PEN test, a physical person coming and testing your system is still done. So I don't think there's any way to get around that. I think if anything, it makes even more important -- lived experience, versus just trained experience. Because there's probably certain intangibles that you can't learn from training that you just experience from living.

Ben Ogilvie:

One more and then we need to wrap it up so we can send our captioner and interpreter home for the day. For the HBER study, was there any expert study knowledge for training, as far as you know?

Michael Bervell: That study, no. But there was a study published five months ago with Procter and Gamble, a similar study, where they did provide training.

Ben Ogilvie: One more in room. Yep.

Audience Member: Thanks so much for your talk. I was just wondering if you had any thoughts on what we would need to change about education so that kids that are growing up with AI are going to become cyborgs and not... Where they're actually mutually co-evolving and amplifying each other's capacities, versus atrophying, potentially.

Michael Bervell: So one thing I do on a weekly basis is I teach high school kids how to make startups. And over the course of a 12-week program -- and usually the first

6 weeks are your startup idea and the last six weeks are vibe coding. Learning how to use Claude by Anthropic and Codium and Lovable to make applications. One of the things is I buy the most premium versions of these tools for them to use and they all share a login. I bring this up, because sometimes I'll go back and... I wonder what they're prompting. At this point, the output is not really the homework. The homework is... How do they do it? And I use that when I go back to teach. So I think it's really interesting to see how high school students today are using these tools. Because they're using it like... In the cyborg way naturally. Where they're not just asking it a question and taking it at face value. It's funny, because obviously I just give them access to this tool. Which they use for the internship. And then some kids also used it for all of their homework. Which I'm like... Should I feel guilty that I've just completely hacked this child's learning and development? And just... They're no longer going to write an essay? Yeah. Or the homework needs to change. But it was funny. Because I was looking... This was even just two days ago. I was looking at... What is this chat, this essay about Odysseus? And it was one of my former students. From six months ago. I was like... Yo. Stop using this. From six months ago. And she was asking a question. She was like... Hey. This is my essay. What do you think I need to improve about it? And it gave her all this feedback. And she was like... Okay. I like these three points in the introduction. Change it in this way. She was co-writing with it. Hey, you lost my voice and my style. Take the style of paragraph 2 and 3 and match it to the introduction.

Is that cheating? Is it learning? Or is it just the way that you have to use tools nowadays? I was actually quite impressed. I was like... Man, I'm such a good teacher. She's going to get away with this. They're never going to know. But that was kind of what I see happening. So I think naturally the way that you use these tools... At least in day-to-day life... Is probably the way you might want to use these tools in our very specific kind of accessibility work.