

Transcript for [What Matters. Episode 26: Getting into React Native with Elias Nahum](#)

Voiceover

Welcome to What Matters, a podcast from the folks at Mattermost. We'll be discussing ChatOps, open source, DevOps, and everything that matters most to you. Let's see what we're chatting about in this episode.

PJ Hagerty:

Hey everyone, and welcome to What Matters, a podcast from the folks here at Mattermost. My name is PJ Hagerty, your host as always. In this episode, we have something pretty special. We're going to have two of our developers, Miguel and Elias discussing React and kind of how they get started and how you could get started. So let's listen in on their conversation, see what matters to them about working with React.

Elias:

The reason that I wanted to chat with you about this is because I know you've been working on React Native for a while now. How long has it been?

Miguel:

I think since 2015 or 16. So-

Elias:

Right, so that would make it-

Miguel:

... perhaps six years.

Elias:

... five years?

Miguel:

Yeah. Yeah. Something like that. Yeah. It's been a while.

Elias:

It's been almost the same for me. Around, let me see, I think it's a little bit less. I think for me it's like from-

Miguel:

No, maybe it's 16, 17.

Elias:

Yeah.

Miguel:

Yeah. My memory's a little foggy, so much going on back then.

Elias:

Yeah. That's sort of my timeframe like 16, 17. But do you recall what made you dive into React Native or how did you become a React Native engineer?

Miguel:

Yeah. Yeah. Like a lot of things in, I would say, my career, was kind of like what was needed of me at the time. So I was playing, I think it was 14, 15, 2014, 15, with Ionic because one of our clients wanted to move a web app that we wrote for inventory management and put it on their tablets. And so I was playing with Ionic, I was playing with Bluetooth modules, things like that. It sort of worked out, wasn't really scalable and what ended up happening with that company is we got

bought out and another team took over that project to scale it out. And then they mentioned React Native and then Scala, and a bunch of other things. They had big plans to redo this whole system.

Miguel:

And the React Native really caught my eye because I remember struggling with Ionic. It was just not really performing, it felt a little sluggish, and so I kind of just started diving into that. And pretty much every prototype app that I wrote after that was React Native. And it was all in the inventory management kind of RFID space. So I kept playing with the Bluetooth libraries and just getting a feel for native modules at that time and I just never looked back from them. It was so easy to spin up a mobile prototype, both in Android and iOS, I was just like pushing them out one by one by one. And in that year that I was playing with them, it was really enjoyable.

Elias:

Right. Cool. Yeah, I remember. So truth to be told, like around 2013, I started this project with a friend of mine. And this project was really about building an app recommendation engine. So basically it was this weird engine that was able to tell you, "Hey, you should try this app or use this one," blah, blah, blah, based on a lot of criteria. But interestingly enough, back then, I didn't want to do anything with mobile development, not at all.

Miguel:

Why is that?

Elias:

So I was basically in charge of the web application and obviously all the backend stuff, that was me, that was me in a nutshell. And after I started working here at Mattermost, that was me as well, web and backend. And I remember we used to have this mobile app which basically was a shim, like just a web view and we saw whatever the web app was at the time. And so I remember the CTO, Corey, he came to me and said, "Hey, do you want to give this thing a try?" And I was like, "No, that I ain't for me my friend. This is not me. I know nothing about mobile, nothing at all." But obviously, I'm not shying away from a challenge. For some reason [crosstalk 00:05:29]-

Miguel:

Bring it on.

Elias:

Yeah. Yeah. It's like, let's just do this. And so I started with a proof of concept and I just loved it, it was so easy. It was basically applying the same concepts that you have on React. Obviously, after a while and after being there for a while, I totally understand that there are a few differences because mobile and web are definitely not the same thing. But it was super easy, the transitions between one and the other. And I was amazed how writing a few lines of code and a few of React components, and then all of a sudden, you have an Android running Native app and an iOS Native app. In terms of how they render, there are a few other things that are not the same. So, in a way, I got hooked, right?

Miguel:

Yeah. I remember that feeling.

Elias:

And it's been so long now but it's something that I still enjoy. And I don't know about you... Actually, let me ask you, did you try Native before Native mobile development?

Miguel:

I did like the tutorials, just touched the surface of it. I went on the Android website and played around, but nothing like production level, nothing really even prototype, never work-related. It was just whenever I had a chance, I dabbled in that, like in many other things, but I never really understood Native at all until actually maybe more recently with Mattermost and diving really deep into the Native side. What about you? Did you write some android app?

Elias:

No, not at all. As I said, I was chasing away from it. No, like running away from it. Sorry.

Miguel:

Yeah. It was chasing you.

Elias:

Yeah. Yeah. It was chasing me, but I was like, "No—"

Miguel:

I can just see the little Android robot waddling right behind you.

Elias:

But after I got in with React Native and everything, as I said before, there were some things that I needed to get done, there was some sort of like required to dive in a little bit into Native. And I can say up to today, until today, I am no Native mobile developer at all. I do follow the code, I do understand Swift and Objective-C and I do understand Kotlin and Java and I can obviously make something out of it and basically come up with the solutions to the problems that we face. But, again, I wouldn't say I'm a Native developer or Native mobile developer, but I find that it's not really a requirement when you are using something like React Native.

Miguel:

Yeah. Yeah. You feel like you wouldn't be able to just spin up Xcode and write like a pure Native iOS app or like Android Studio and just... I couldn't do that either, like activities and all that. I think I could if I were to now take a tutorial and do a little Hello World app. It'll take me like a fraction of the time it would maybe without the React Native experience just because I understand the language and how Xcode and Android Studio work. But definitely, I don't think I'm a Native developer in that sense. But like you said, it's not a requirement.

Elias:

Yeah. It's not. But you brought up a really good point. Like today, if I'll go look into this Swift UI thing and dive into more Android Native development world, which I wouldn't, because I'd need to do the same thing twice, that's why I wouldn't. But on the other hand, it's like, if I would go and do it today, I think with all the tools and the knowledge that React Native has provided me, it would be even easier because now I know what you're talking about. I understand the complexities of the network, I understand what the view is and what an activity is and all that stuff. Like fragments and all those things. Especially things like the gesture system, which for me, before, it was either nonexistent or maybe a mouse hovering on top of an element.

Miguel:

Right.

Elias:

That was it, right?

Miguel:

Yeah.

Elias:

And if you had some desktop experience as well, nothing like the gesture system was a thing, it wasn't a thing. And so now, just because of the nature of this thing, I can definitely understand it. I started with [inaudible 00:11:26] and stuff, and then the VR library that is massive out there, which is the React Native gesture handler that gave me a lot of tools and experience to fully understand what it means. So, if I would've gone back now to Native, I think I know what I'm doing. But again, why would I go in if I don't need it, because React Native, with that, I can build a good enough, very close to Native experience.

Miguel:

Yeah, that's the beautiful thing about it. Actually, we're just chatting a little bit about Native, but one of the things that I just thought of right now is how you had, on the React side, you had React experience coming into React Native. I actually learned React through React Native because I was playing with Ionic on the web prior to that.

Elias:

Oh, interesting.

Miguel:

How was it for you knowing React already, and then coming in and then learning or applying React? Did you feel a little weird?

Elias:

Not at all. It was super simple because on web you have things like a dev and so obviously in React Native, you don't have something like that, you have a view, right?

Miguel:

Yeah.

Elias:

And today, the documentation of React Native website is incredible compared to where I started. But when I started, it was there, it was so simple. They explained it so easy back then, that for me it was like, "Okay, so this is the replacement of this thing, and then the life cycle is the same, and then props are the same." Basically, it's the same thing. So for me, it was super, super easy. Again, there were a few things that if you didn't know, you referred back to the documentation there, which was at the time, if you compare that, it wasn't great like it wasn't. Today, it isn't great either, but it is way better than what it used to be.

Miguel:

For sure.

Elias:

And you go through it and it's really simple once you... The more time you spend on it, the more you understand it. There are things that I still don't understand to this day like measurements, when they talk about units and stuff, it's something that I get lost. I sort of understand what it means, and after a while that I haven't used it-

Miguel:

It just goes away.

Elias:

... like a month or two, it's like, "What was that?" But again, super easy. It basically grabs the React component that you have over here, even the same logic and everything, [inaudible 00:15:07]. Back then, I remember we were using flux and we were transitioning to Redux. Actually, I made the transition to Redux and so I needed to understand that part as well. But it was easy because it was all JavaScript and it was all React, something I was already familiar with.

Miguel:

Nice. Yeah. Yeah. I was with an Ionic, so I was just getting the hang of Angular at that point. But I knew that Ionic wasn't cutting it for mobile, and at least not back then. And so, the team that took over the code that I had written in Ionic, they were pushing React Native on themselves. I still talk to one of the guys that joined that team and he was just like, "This is great. This is great. It's so much easier." And I was like, "Oh, now I have to understand this whole thing. And then with state management, I have to understand Redux." There was a little bit of resistance for me initially because I felt like I was picking up the new kid on the block all the time. And I was like, "I need to focus and really get good at something."

Miguel:

But yeah, I gave it about a month, a month and a half where I just dived deep into React and did a few of the web app tutorials and kind of saw the similarities there and like how easy it was just transitioning into mobile with React Native. And actually the first it wasn't a production app but it was a startup app, so we were pushing it on some clients. We were trying to do like a social network type thing around security cameras. So, like if you have your own IP... Everybody has IP cameras, different brands at their home so what if we could connect this into a hub, invite your neighbors, invite your family members to watch your streams, and then the community can kind of keep watch, keep a neighborhood watch on each other, something like that. And so there was a web app component, but there was also mobile because you want to see these things when you're away from the home, right?

Elias:

Yeah, obviously.

Miguel:

You want to see your streams. So it was super E like it was so easy to transition from the web app and fix a bug there and then go back to the mobile app and do something there, and it felt so good to be able to do that back and forth. So I think that's where I kind of realized there's a lot of value here, especially if you have a web app and you want to either transition that to the mobile or maybe even start with the mobile and you want some kind of web interface to that too, it's the same. I kind of understood the motto, what is it? Learn once write everywhere? That really hit me then. It's really, really true. So like anybody [crosstalk]-

Elias:

But it could be better, it could be better.

Miguel:

In which way?

Elias:

It definitely could be better because I know there's help with React Native Web and Reactive Native Windows and React Native Desktop and whatnot, but can it be better? Can it be not only "learn once write everywhere," but reuse all that you can possibly reuse and just change things about certain parts of the logic? Because one of the things that I found is we're building this thing on what? And it was on React, and yes, we use Redux and all that. So one of the things that we attempted at the time, and you were part of it, like grab everything Redux, put it in a different library, and share it between the codebases, which-

Miguel:

Worked for a while.

Elias:

... worked for a certain extent. But the thing is, obviously there are certain things that the web can forgive, but mobile, it just won't. You're going to have performance impact, you're going to have, "Why would you run the same logic and make all this request when you don't need it." Because in a way, even though we have incredibly powerful devices today, they aren't a match for a computer, they just aren't a match. So as you squeeze your computer more, and you share that same, let's say components, logic or whatever between the two, you're killing the small device, you're just killing it.

Miguel:

I think actually that brings up one of the appeals if you're a web app developer and like maybe picking up React Native, like you said, developing for the web can be forgiving because obviously there's an abundance of resources like memory and CPU, but when you get to the mobile world, you're limited there. And that limitation forces you, depending on what kind of code you're writing, maybe you won't hit any performance issues at all. But at least at Mattermost, for what we're doing, we did hit that. And then you got to rethink, you got to start thinking more efficiently and look at the bottlenecks and really analyze this stuff. And it's fun stuff to do, but I think it's a good lesson to learn as well.

Elias:

Yeah. And again, you can get all those practices that you've learned just because of restraining yourself in a way. And when I say restraining it's like, "Really, are you restraining yourself or are you just finding better and more efficient ways of doing things?" And you can benefit from that and bring that knowledge to your web app so that web app becomes even better, and not the other way around. Because when you do it the other way around, it's like, "Okay, let's start with the web app, let's bring it to the mobile app... What's going on here? It was working perfectly over here." But when you do it the other way around, you start with a mobile-first approach, it's like, "This is working great, put it over here." It's like, "Ta-da! Whoa, this is amazing," or whatever it is. I've been there, I've faced that a couple of times. But yeah, it's pretty interesting.

Elias:

But the one thing that I want to bring back, so we can talk a little bit more about this is, well, I can tell you myself because I've been both a web developer and mobile with React Native, I can tell you my sort of experiences in what I find different. But what about you? Because I know you were working on Ionic, but Ionic is basically HTML and basically you were building a web app for a phone. So what did you find you needed to do a little bit different or massively different, I don't know. But what are those things that you found that, "Yeah, if I do this thing, I wouldn't done it for a web application. I'm hanging myself."

Miguel:

Yeah. Well, one of the things that we did in that app was... So what we were comparing was, so you have... Let me backtrack and just explain a little bit of the project. So it's a company that sells jeans. I don't know if I can say the name, but it's a company that sells jeans. They're well known here in the States and they want an inventory tracking system. So you get all the jeans, you put tags on them, RFID tags, they have the UPC encoded on them and that within itself is encoded the color, the size, if it's men's, whatever. And then on the ceilings, we had RFID antennas that are constantly blasting RFID. They power up the tag, the tag then sends back its EPC. It gets decoded, and eventually, our system got...

Miguel:

Well, our system did a little bit of the decoding too, but the end result is that we had a database with a location of a particular type of jean, a particular size and all that. And so we can then compare what's actually being read live to what their legacy inventory management system had because they wanted to move away from that. And so the first thing that we were doing was just comparing the efficiency of our system. So we had a bunch of graphs showing how our reads compared to their reads and we were using D3, I believe, at that time, and the web app was really easy to just show all these graphs. And as your reading tags we fed into that system, it was very reactive and you can see the graphs moving up and down the bar graphs, stuff like that. And it was a really cool, nice little wow factor.

Miguel:

But once we got into the tablet world, we were using Android tablets and we were using ionic there, the thing would just freeze and crash and the animations were just really choppy and it just really wasn't working. So I think that's one of the things that I definitely would've done differently, looking back. This is not the web, these are really cheap Android tablets too, this is like bottom line. These things are sluggish and how do you give a good experience with that limitation? It's something that I didn't really think about back then, I was just like, "Oh, I have the code here, and I just put it on here and ta-da! Here you go, it's a mobile app."

Elias:

So for me, one of the biggest ones, and I mean the biggest ones is when you are working on web, you don't really think about persistence, everything is just loaded on demand. Which is fine, you're a web browser, it's sort of expected. But when you have a mobile app, you open the thing and you don't want it to say, "Loading, loading, loading." Loading here, loading there. And more than that, it's not the fact that it's loading because eventually it needs to load data, and then persisted or whatever, and then provide like offline support or everything like that. But also, the fact that if you want to loathe the same thing over and over again, there are people that are on cellular networks that have a limitation on their plan, they came for that. So that was something that at the beginning was just like you, right?

Miguel:

Yes. Like you don't think about it.

Elias:

"Oh, yes. This is great, easy," whatever. "Let's bring it in. Boom and everything was loading and loading."

Miguel:

Yeah, exactly. Those are the two... A couple of things that you have to be mindful of. One is animations, obviously, but then also network requests, there is a cost to those things on the mobile app. You not only make them and then once you get the data, memory limits you to how much data you can have, there too as well. So it's-

Elias:

Loading it at the same time.

Miguel:

... something you don't think about.

Elias:

Yeah, loading it at the same time. So yeah, that's an R factor, how much data can I have on memory at the same time while running the thing? Another thing that I want to mention here is, this is really, really funny in my opinion, so you know how on a web application, when there's no internet connection, you get the browsers whatever screen that they show up?

Miguel:

Yeah.

Elias:

Some of them have games, some of them have other stuff.

Elias:

Which it is great, right?

Miguel:

Yeah.

Elias:

You cannot bring that mindset to the mobile application. You need to correct those gracefully. You need to be the one creating the game for the person not to get frustrated. So, for me, that's a very funny one because when you're coming from web that's not something that you consider. And you know what? It doesn't happen only to developers or engineers, it happens to everyone. Like designers do not take that into account. They do after they've been there or they got a bit of experience, but not before that. It's like, "Whoops, what do we do in this case?" So for me that's super funny.

Miguel:

Yeah. Time to launch, time to load, those are things... And we're talking about these things that as mobile developers we deal with. But just bringing it back to write once, sorry, learn once write everywhere, the cool thing about React Native is that you get to deal with that without any... You don't have to figure it out on the Native side. Maybe you do, maybe you would have to dive in there, but if you're a web developer and you know React, you come and apply React here, and you don't have to think about anything else other than, "Okay, let's solve this time to launch issue." It might involve a little tweaking on the React Native side, but you have all that wealth of React information and you don't have to relearn something new. So you can focus-

Elias:

Yeah. Absolutely.

Miguel:

– on the animation. You can focus on the network request, you can focus on the time to load. And it's more business logic. So it's your business logic that you're now really attentive to and not the boilerplate, not anything like that.

Elias:

Yeah. Yeah. I absolutely agree. So to close this off and some other day we can talk about something else, but look to close this, if you needed to score, like to score from one to 10, how easy it is for a Native mobile developer to switch to React Native, how much would you give it?

Miguel:

That's a tough one because I know some native developers, but if they're Android, they just stick to Android, they don't really care about iOS or vice versa. So I don't know what that experience would be like. My guess is that it would be a little bit more difficult because they're used to developing a particular way for say Android, and now they're being asked to change that kind of paradigm and move a little bit more on the JavaScript side of things. And it could be one of those things where if they struggle with a particular bottleneck in React Native, like let's say an animation, or maybe like a network request, maybe I'm just guessing, they might have that back in the head feeling like, "I could just do all this in Native instead." So maybe that's a struggle.

Elias:

I agree with you. I agree with you to a certain extent because, okay, so you mentioned someone like an Android developer, right?

Miguel:

Yeah.

Elias:

My best guess is that they will suffer from frustration at times. But then they turn their chair over here and say, "Oh, but I have it right on iOS too." And it's like, "How long would it take you to do it on iOS and how many hours of frustration, because you don't know how to do it, you'll suffer because of it. And so it sort of like here and there, so I would give it around a seven.

Miguel:

Okay. Okay. Just on a little tangent there, you know how they have those videos where people react to things? Maybe we can do like Android developer reacts to React Native developer. And he's like, "Why is he doing that?"

Elias:

Hey, that could be a fun episode. Like absolutely, yeah, that would be great. And so same question, how easy would it be for, let's narrow it down, a React web developer to jump in and start working with Reactive Native and do something to the same professional level that they do it on.

Miguel:

Right. I think it depends. Let's say it's an app that doesn't have animations or very little, you're not loading a lot of data in the beginning, none of these kind of bottleneck things that we talked about earlier. It's a 10, like it's super easy. You'll spin up an app, just like... Obviously, there is setting up your environment and all the dependencies and all that, but pushing that aside—

Elias:

But web has so many freaking dependencies. There's so many different environments that it's just another day in the office. It's the same thing.

Miguel:

Right, right. So, it's surprising how fast it is to just push out a mobile app. And then once you get into, again, the bottlenecks, you just have to be mindful of that and it's just something you kind of learn along the way. But it's not anything terrible or scary, it's challenging, it's hard to get right sometimes. Not because it's a difficult problem, but I think it's because, again, it's always a trade-off between speed and pretty much... Yeah. It's a trade-off, like anything else.

Elias:

Yeah. I would agree. For a React developer, you already know so much that using React Native would be super easy. And then, yeah, having those things. So I would say with the bottlenecks and everything in between, everything about animations and something that we didn't discuss here about the bridge and all that stuff, I would give it like a 9.8. Not a straight ten, a 9.8.

Miguel:

Yeah, 9.8.

Elias:

Right. So basically if you are React developer and working on web, React Native might be your way to get into mobile development like this [crosstalk 00:35:57].

Miguel:

Yeah. And to be more clear, it's not just like a component, it's the whole ecosystem in React. Like if you know Redux for state management or MobX or even just using context, it translates to React Native. As far as like network requests, there's Fetch, so you don't have to do much, it's all there. So it's really all-encompassing. You can do everything you can do on the web app for the most part on mobile. You can just port your project over and you have a mobile app for it.

Elias:

I agree. Well, Miguel, thank you. Thank you for this. Thank you for the conversation.

PJ Hagerty:

Yes, Miguel and Elias, thank you both. We're really glad you had this conversation. We hope that it's something that matters to a lot of our listeners. Thanks again for listening to What Matters, the podcast from the folks at Mattermost. If you'd like to join the conversation, jump into community.mattermost.com. You can find me on [pj.hagerty](https://twitter.com/pj.hagerty). Elias's also on there, [elias.nahum](https://twitter.com/elias.nahum), N-A-H-U-M, and come join the conversation. Let us know what you think matters most.