##### [00:00:00.060] - Introduction

Welcome to EU Code Week podcasts, we bring coding, computational thinking, robotics and innovation closer to you, your community and your school.

##### [00:00:17.710] - Eugenia Casariego

Hello, everyone, and welcome where your host Eugenia Diego and Maha Elkheir.

##### [00:00:22.060] - Maha Elkheir

With this series of podcasts we founded EU Code Week Tim would like to contribute to changing the education system in Europe and adapting it into a society that is getting more and more digitalized.

##### [00:00:32.590] - Eugenia Casariego

Maha and I are part of the EU Code Week team supporting our lively community of educators to teach coding and computational thinking. In this episode, we're going to be looking at how coding is, at its core, a creative activity. We will be discussing today about the work of Code.org, a US based non-profit dedicated to bringing computer science to all young people, regardless of age, gender or background. They strive for computer science to be part of the school curriculum with a vision so close to Code Week's,

##### [00:00:58.030] - Eugenia Casariego

We have cooperated closely with Code.org, including sessions on their CS fundamentals curriculum in our teacher training courses. And in fact, this year they have designed three of our challenges. We invite you today to explore this topic with us.

##### [00:01:09.700] - Maha Elkheir

Exactly, and we are excited to speak about that with our guest from today Kenneth Akiha, a curriculum development manager from Code.org. So welcome Ken, and thank you for joining us. Can you introduce yourself to our listeners and tell us a bit about yourself and your experience?

##### [00:01:25.360] - Ken Akiha

Yeah. Hi, it's great to be here. My name is Ken Akiha, like you said. I been working for code.org for the past year. I'm based in the Seattle area, in Washington state, in the United States. Before joining code.org, I worked as a classroom teacher, both at the kind of grades six through 12 level teaching both science and computer science. I also did some work as a facilitator, training teachers, how to implement computer science into their classrooms.

##### [00:01:55.420] - Ken Akiha

And yeah, and then this past year, working at code.org. Like you said, I work on the curriculum development. And so that means making things from lesson plans to teacher resources to collecting feedback from teachers and students about how our curriculum is working for them in their classrooms.

##### [00:02:13.210] - Maha Elkheir

Great. As you can see, you're very passionate about making computer science more prominent in schools and in your case in the US. Why is that and what inspired you to advocate for teaching the subject in schools?

##### [00:02:24.490] - Ken Akiha

That's a great question and definitely gets at the heart of the mission of code.org as an organization. And for me personally, the reason that I got into teaching in the first place was kind of equity at the base of it, realizing that just as it is in other parts of the world, but certainly here in the US, a lot about the access to education was determined by basically your zip code, where you, where you grew up, where you happen to be living.

##### [00:02:50.080] - Ken Akiha

And to me, that doesn't seem right. And I think also what drew me to computer science was the experience with the students that I taught it to. At first, I saw how I was able to connect with them in a different way than I had found with other subjects. I think mostly because it was so relatable to their lives, and they felt that what they were learning and what they were creating, they had a direct application to to problems that they saw or ideas that they had or things that they wanted to create.

##### [00:03:19.150] - Ken Akiha

And so that connection was really strong and it gave me a lot of energy as an educator. And so wanting to to spread that type of experience to other students and teachers as part of my work now.

##### [00:03:30.460] - Eugenia Casariego

Yeah, that makes a lot of sense. And so I have a couple of questions. So you currently work with the curriculum development at code.org. So I have a couple of questions about these curriculum and CS fundamentals curriculum. So I'm wondering, first of all, you've mentioned equity and you mentioned that this is something important to you. So how would you explain to our listeners, how does CS fundamentals curriculum and code.org in general promote equity across the classrooms and as well,

##### [00:03:54.430] - Eugenia Casariego

How does this curriculum transition from primary to secondary school?

##### [00:03:58.480] - Ken Akiha

So I would say that code.org equity is one of was one of our core values, and the way that that comes through is in a few different ways. So I think first acknowledging that as a curriculum team work when we sit down to write down any lesson or any new course or anything like that, the teachers that we have in mind that we're trying to support and the students that those teachers work with are the students that we currently don't see represented in tech industry and computer science related jobs.

##### [00:04:26.980] - Ken Akiha

So kind of primarily those would be female students and then also students from traditionally underrepresented groups. And so in terms of how that comes across in the in the classroom, I think traditionally the biggest thing is just been to try to, like I said, make access as easy as possible for anyone to bring it to their school. And so really thinking about when we design for thinking about the teacher, thinking about a teacher who might not have any experience with computer science prior to this, right?

##### [00:04:55.180] - Ken Akiha

And so what can we do to support that teacher in terms of things that are put in the lesson plan, things that are put into the lesson itself that help support them, feel confident in implementing a lesson and delivering it? And then also within what we what we make a lesson in terms of who is represented. And so we have a number of videos that go with our curriculum and trying to think about who are the faces that are showing those.

##### [00:05:16.740] - Ken Akiha

So when students are seeing that, who is it that they are seeing as a computer scientist and what are they, what's that image that they start to develop of what that means and what type of person is that? And is that someone that they can identify with and see themselves in? You also asked about primary to secondary. So currently we have three, I guess, courses that we call them. So computer science fundamentals is it's kind of for grades K through five and then computer science discoveries is, say, six or 10.

##### [00:05:44.520] - Ken Akiha

And then our computer science principles is for high school grades. And I would say that the way that the kind of transition works is if you think about it from high school and computer science principles is really mostly implemented as a full course kind of standalone computer science course. It's actually connected to the advanced placement program that we have here and so students can earn credit through that. But that's one reason why it's often implemented as a full course. But as you go down towards the primary grades, it becomes more of a of a modular implementation.

##### [00:06:19.560] - Ken Akiha

So even computer science discoveries can be taught as a as a full course. But we find that lots of teachers are using individual units throughout that, and we designed with that in mind, knowing that you might not do every unit in the course. And then certainly with computer science fundamentals we actually have within computer science fundamentals, we actually have courses, A through F. So even within that, there are a bunch of small, different courses in that I would say even more so in the elementary primary grades.

##### [00:06:46.860] - Ken Akiha

We have teachers who implement it in all sorts of ways. So whereas middle school and high school, it's more common that it's an elective or kind of a standalone computer science course in primary grades, it's more often or it's more of a mix. So sometimes it might be more integrated into other subjects or maybe taught as its own thing. But it's not. It's not as much kind of on its own. And in terms of like you kind of mentioned, like, does it get harder or not?

##### [00:07:11.100] - Ken Akiha

I think the subject matter is different, and we certainly, yeah, we're writing for a different audience. But in terms of the computer science content, even computer science principles doesn't assume that a student had any computer science background prior to entering that course. And so we try to lower that barrier for access no matter where it is, because we acknowledge that students at this point, we have a vision that it's not like this forever. But at this point, the reality is right and I can speak from my own experience is that 90 percent of the students that I taught when I taught them computer science principles hadn't taken any computer science before.

##### [00:07:43.020] - Ken Akiha

And so we were maybe working with some things that required some yes, assumed a high school audience right? In terms of like what you know, what they're doing in terms of the writing and reading background. But in terms of the computer science, it really was starting from a baseline of no experience.

##### [00:07:59.550] - Eugenia Casariego

Right. And so to follow up on that. So in Europe, we tend to have an approach to computer sciences a bit more cross curricular rather than dedicating a whole subject to that. So what do you think is best to tackle this lack of background in computer science? Do you think it's best to include a subject as a specifically computer science, which is, for example, done in Ireland? Or do you think it's best to include computer science a bit more cross curricular, across different subjects?

##### [00:08:24.900] - Ken Akiha

I wouldn't say from our perspective that there's there's a one that's better than another. Like I said, traditionally we as an organization, we have focused more on the standalone kind of computer science as its own thing approach. But we acknowledge we're at a point now where we we are starting to do more work with yeah, cross curricular implementation, realizing that that for a lot of places, even here in the US, that's that's the approach that is going to work best for them and it's going to be the way to help the students in those areas.

##### [00:08:54.000] - Maha Elkheir

Great. Thanks a lot. So drawing on all the things that we have discussed and on the challenges that you have mentioned, that is equity and access. How do you use coding and computing in your class? Like perhaps you could share some concrete example? What is your day to day classes?

##### [00:09:09.060] - Ken Akiha

One big way that we try to implement coding in our classroom is through what we call unplugged activities. And so these activities would be things where students are maybe using physical, physical manipulatives like even something like sticky notes or plastic bags to to try to understand how variables work. But they're not actually using a computer, but they're using those concepts and they're talking, starting to develop a mental model for those concepts. Lots of different unplugged activities we find again, those help with equity, too, because thinking about students who might come into a course with different levels of background, those unplugged activities we find usually kind of level the playing field in terms of thinking about coming at things from a shared experience and so that everybody's on the same page.

##### [00:09:52.320] - Ken Akiha

No one is kind of advantaged by maybe some prior experience or whatever they might be coming into the course with. We also find that those really help build community and kind of engage students in a different way than when they're on the computer. Another way that we use coding is by having students read, you know, code that was written by somebody else. And so looking at pre-written code, trying to understand how it works. Discussing it with themselves, with, you know, with their classmates, discussing it, with their teacher, asking questions about how it might be working, you're trying to test it out and see what they can discover about it.

##### [00:10:27.000] - Eugenia Casariego

And you have mentioned now a few ways in which we can creatively bring computer science into our classroom. You have mentioned that kind of project based learning, if I can call it that. Playful learning as well and different approaches. And I was just wondering about the different subjects, as traditionally in school, we have very different boxes where we put kind educatic materials not use-. You taught science, for example, so that was one box. So now I was wondering, so for some subjects, is it easier than for others to integrate coding and computational thinking?

##### [00:10:57.810] - Eugenia Casariego

And how can we make it easier for those subjects that we traditionally may not bring coding to?

##### [00:11:02.910] - Ken Akiha

Yeah, that's a great question. It's definitely something where we're thinking about and trying to grapple with right now. At this point, I wouldn't say that I have a clear feeling opinion that one is easier than the other because I think we have examples where it can work well for kind of all different subjects. If I think about who our teachers are, they're certainly, at this point, there are more teachers who I would say have more of a math, science background who then have kind of maybe started teaching computer science.

##### [00:11:33.310] - Ken Akiha

So I think from that perspective, in some ways you might say, you know, from the teacher background perspective, there's there's maybe a tendency to integrate it with math and science. But that being said, I think there's a lot of factors that are contributing to to why that is. And I would say even some stuff that we're trying to fight explicitly fight against that. You have to be good at math and science. To be a computer scientist is something that we're trying to directly combat.

##### [00:11:59.400] - Ken Akiha

So I think what we're trying to do right now, and I think what would help to bridge the gap, to two things I think. Just show some examples of what it can look like. I think for a lot of teachers, there's a desire to integrate some computer science, but there's not a clear image or not a clear understanding of what that looks like for their class. Maybe they just have never seen something like that. So I think providing some examples of we're working on something right now with poetry and coding, so I can imagine a teacher seeing that and seeing as a as an English teacher and saying like, Oh, OK, I never even thought about that.

##### [00:12:32.730] - Ken Akiha

But now that gives me some ideas about how I might do that exact thing in my class or something related like that in my class. The other thing, what we're trying to help in terms of bridging that is just different levels of integration. Basically, all the way from, you know, pretty heavily involved, almost like a full unit or a full what we're calling modules where there's it's kind of all scripted out for you. You can kind of take it and use it, but it's pretty.

##### [00:12:55.590] - Ken Akiha

It's it's a high involvement in terms of your classroom time and maybe the preparation for a teacher. All the way down to like one just one lesson. Right. So one thing that you might just try because it's related to something that you're doing and in your in your math class or your history class that week. And then some things in between something that we're trying to work on to is like kind of these more open-ended projects, so similar, like give some examples, but, and provide a framework, but not as much of like a scripted lesson plan, but some ideas that teachers can then take and then adapt to their own classroom.

##### [00:13:26.400] - Ken Akiha

Because we realize at this point, I don't think there's ever a future where we're going to be designing something that's aligned with everything for every different subject, right? There's just such a wide variety. And so it's more about how do we give teachers examples of, and the skill and the confidence? Then they can take it and adapt it to their own whatever they might be teaching with their students?

##### [00:13:44.850] - Eugenia Casariego

Yeah, I think I can agree with that. It's it's not about just telling them, Okay, do this and do that, it's more about suggesting and kind of providing that kind of support or scaffolding, if we can call it that? I think from Code Week as well, we can subscribe to that opinion in the sense that we do believe that coding and computational thinking or computer science can be integrated in all subjects. And I think it's a good moment to remind our listeners that at codeweek.eu you can find resources to integrate coding and computational thinking and computer science across the curriculum, and you're going to find a lot of materials

##### [00:14:15.390] - Eugenia Casariego

that will help you to do so. And you'll find a lot of support to do so. But I want to get into another point and I were talking about the why and we're talking about the how. And so we've covered and we've discussed about the obstacles in access for students. So we talk about equity, we've talk about difficulty in access. But what are they in your opinion? What are the biggest obstacles for teachers and how can we make them easier for them to bring computer science or computing to their classroom?

##### [00:14:40.530] - Ken Akiha

From my perspective, the biggest obstacle for teachers is time. I think time, both in their schedule, what they're teaching. And again, I'm speaking from kind of a US perspective, so I'm not sure how, if it's if it plays differently in Europe. But you know, most teachers that I worked with and that I work with now are already pretty overworked and overscheduled and overbooked. And so asking them to to add something new was just kind of, you know, could also just kind of felt like it could be the straw that breaks the camel's back or just very easy to say no to because

##### [00:15:16.470] - Ken Akiha

Already felt like were doing all they could, and maybe you weren't feeling supported, so I think what we try to do with our approach is not to say that it's not. There's the curriculum piece, but we also offer a professional learning side with all of our courses. The goal there being that we provide teachers with with a community of support. So because I think I would say maybe that's another obstacle, especially in lots of areas and in the US, you know, you might be the only person trying to teach computer science in your whole school or your whole district, your whole town.

##### [00:15:45.240] - Ken Akiha

And that can be kind of an isolating feeling and a definitely an obstacle to persisting with with it. So I think supporting teachers with that community, you know, the professional learning that we offer is for the computer science discoveries in the computer science principles as a five day workshop in the summer and then four weekends for single day workshops throughout the year. Yes, I think just going back to the first point that I made about time too, I think we try to value like the time that a teacher has.

##### [00:16:16.830] - Ken Akiha

So knowing that if they are going to attend professional learning, right, how are we making sure that that time is the most valuable and if they are picking up our curriculum to use it right? How can we make sure that we have provided them with the most supports and resources in those lesson plans and in the in on our platform that they that they feel like they can figure it out and make it useful for their situation?

##### [00:16:42.120] - Maha Elkheir

Thanks a lot. I think it's really important, what you mention about time, but I think that's an issue that almost I think all the teachers have. But one important point that you also mentioned is the sense of belonging and the sense of community in this is actually what we try to do as well in code week, to create the sort of, I would say, family for them to be able to to reach out to other teachers, maybe even in the same country.

##### [00:17:04.290] - Maha Elkheir

And to see that, actually, we might have the same issues, in my experience, the same problems and with that they can help create some sort of action plan all together to solve this issue. So I think it's it's actually really important because a lot of teachers, for instance, might teach in some rural areas, where they feel isolated, like you mentioned. And so I think it's really important to emphasize on the fact that they do have a community and it's super important.

##### [00:17:34.090] - Eugenia Casariego

Code.org encourages the integration of computer science in curriculas, as we have discussed right now, and you have actually been quite successful at that, if not very successful. And in Europe, as I as we were commenting Code Week has steadily been growing since 2013 and involving millions of educators all over the continent and even actually beyond. But Ken, how do you see the future? Do you think we can look forward to national curricula, which incorporates computer science? How do you see that our efforts will lead us five years from now, for example?

##### [00:18:02.980] - Ken Akiha

I wish I could see into the future, for sure. I think speaking for the US, education is something that's very localized here in the US. Definitely like state by state, but even even district by district, right? Decisions that are made and the autonomy that's given to those districts. And I think there's definitely pros and cons to that, for sure. But because of that nature, I think I wouldn't say a national curriculum is something that we we talk a whole lot about here in the US and really think of as kind of as something in the future.

##### [00:18:33.640] - Ken Akiha

We think of it as it's going to be kind of a piece by piece approach, right? And so some some states are going to make it kind of mandatory across the state and others might be kind of more of a bottom up approach where some teachers teachers are picking it up kind of like I did and lots of other folks are doing all the time and kind of spreading it in their districts, spreading it in their their community, in their state. And at the same time, I will say, I think there is momentum building towards what I think a lot of folks who teach, teachers who teach it.

##### [00:19:05.560] - Ken Akiha

It's kind of the reason the something is definitely a part of the reason why I started to teach. It was I mentioned earlier the connection that my students had it about it. And so I think from that perspective, there's a growing momentum nationally here that that it is something that everybody needs to know. The part that's less clear, I think, to me and to us is how exactly that happens. I think that it's it's happening. I'm hopeful and I think I would say I'm even confident that it's going to continue to grow because I think more and more places are saying these things out loud and more and more governments and folks who will kind of control that sort of stuff are talking about computer science out in the public, but I can say that I would like it.

##### [00:19:45.850] - Ken Akiha

Yeah, that just that concept of national curriculum, I think, is something that we don't talk too much. Yeah.

##### [00:19:50.740] - Maha Elkheir

We notice as well that digital technologies such as AI, cybersecurity, they all developed very fast. So considering this, how do you think schools can keep up and how can they how does this affect actually the teaching of the computer science and computing in their classes?

##### [00:20:08.500] - Ken Akiha

Yeah, this is definitely something that we talk a lot about, right? Yeah, it's it's how can schools keep up? How can teachers keep up? How can we, as as people who are writing the curriculum, keep up? Part of our goal with our curriculum is to, is to empower students and also to empower teachers to kind of view the skills and the tools that they're learning as ways that kind of help them discover and learn about the world.

##### [00:20:33.280] - Ken Akiha

And so, you know, this idea of kind of always learning. I think really goes well with this. And just in terms of like the acknowledgement that, yeah, this is different than it was last year, right? This is changing really fast. And I think, you know, I think it's something honestly for for what I've found for students like, that's not something that they struggle with as much as maybe adults, just because they live in that world.

##### [00:20:57.310] - Ken Akiha

A little bit more. Things are always changing and that's that's kind of the world that they've always lived in. And so I would say it's it's more of from what I see of the challenges just to the school side and the teacher side just to feel like, I think two things. So the acknowledgement that we we as a as an organization that's providing curriculum, we're not going to be able to be fully up to date all the time, right?

##### [00:21:19.030] - Ken Akiha

Things are just changing too fast and we don't have the bandwidth to be rewriting every single lesson when something changes. And then along with that, I think we also we also hope that the lessons that we do write, and the skills that we do teach are some of the kind of more like underlying concepts of how things work. And so far, at least our curriculum doesn't get too far into like the technical super up to date most recent methods of doing of doing like a, you know, of a computing language or kind of network security, more of like the underlying principles that are there and that we hope that then that gives our students and then also our teachers like the confidence then to go off and learn more and maybe dive into like what is the current, you know, the most current version of this?

##### [00:22:00.160] - Ken Akiha

And I think that's something that we we hope that folks are empowered to do.

##### [00:22:04.100] - Eugenia Casariego

I just had a bit of a follow-up question to that and so, as you say it is indeed almost impossible to just catch up with how fast things are happening. And so would you that advocate or would you then agree to give room to a more flexible curriculum because right now, at least in Europe, I'm not sure how is in the US. We have quite rigid curriculums with quite structures on the content of its unit, and its lesson. I mean, it varies a bit from country to country, but I think that's the general idea.

##### [00:22:28.270] - Eugenia Casariego

And so would you agree then to have more of a flexible curriculum, do you think that could be a solution? To these fast changes and to assimilating these changes?

##### [00:22:35.730] - Ken Akiha

Absolutely. I think that's it, that's essential because like I said, we're we're not going to stay up to date on every single change, but we definitely take that approach of like a flexible curriculum. And I think that's important to take. Yeah, if we think about that from like a national or even like a state level. I would argue that that's it, that that would be a good approach, no matter the content area. Probably, you know, because I can think of some things that we still teach in in math classes here in the US that I would argue could be updated and could be thought of more flexibly.

##### [00:23:04.410] - Ken Akiha

But I think for a computer science, an area that we've talked about is changing so fast that that's that's essential. You know, I think an example is that when we started making our curriculum, AI wasn't something that was really on our radar. But obviously it is now as it is with everybody. And we, yeah, we made a big investment to make a whole unit about artificial intelligence and machine learning that we just published this past year. And so I think when we see these bigger swells of things that are topics that are becoming, you can't ignore these and they're and they're essential and they're just as important as maybe, you know, learning how the internet works like something that was maybe an older concept, but AI being something that's a little bit newer.

##### [00:23:43.800] - Ken Akiha

We're going to adapt to that and we're going to make curriculum that meets that. But I think it's more of these. So, so more of the like broader kind of still still fast moving changes, but these kind of larger areas, rather than maybe trying to stay up to date on the exact details of internet encryption or whatever that whatever the subject might be.

##### [00:24:04.530] - Maha Elkheir

Yeah, I think I agree as well. I mean, digitalization is happening very fast and there is a gap between the teachers and the students because obviously, with all the new tools that are there already. But this leads me to my second question to you perhaps have a sort of suggestion for this teachers who would actually like to bring, you know, the changes that we talked earlier about, like flexible curriculum or any of changes that you may think of what could be the tools and activities and argument that they can leverage?

##### [00:24:34.350] - Ken Akiha

In terms of the approaches that teachers can take, I think some approaches that we advocate for are at a big scale level, you know, as much data as possible to use. I think that can help get the attention of people who make decisions. So we talk about data in terms of jobs, kind of job. There are lots of jobs open in the US that are that that are looking for folks who have a CS background and kind of the salary of those jobs. We also talk about we have kind of identified as or organization like nine policies that states can adopt to help promote computer science education.

##### [00:25:09.480] - Maha Elkheir

And so there's a portal that we have that where teachers can look at those at their state and see which ones have their states have passed and which ones they have in in that. And that's something that they can like advocate for kind of with their local officials or their local legislatures. And then I would say, like from the more from the bottom side up something like that I have talked with other teachers and witnessed myself is I always think that like students and student work is the most impactful way to show the value of anything you're trying to do with students.

##### [00:25:42.460] - Maha Elkheir

But certainly computer science is no different. I think this can happen in so many different ways, but it might be having students present at a school board meeting about projects that they've done a really common way that, you know, teachers get more students to sign up for their classes is by students recruiting other students, right? And so I think letting the students do the talking about why they think computer science is valuable and why they think it's something that's worth that they enjoy.

##### [00:26:10.560] - Maha Elkheir

And it's it's worth their time is another effective way to help kind of leverage, like why should we be doing this? Why should we? We should we should we be doing more of this than than we have currently? So I think those are that's kind of my top down approach and then my bottom up approach suggestions when it comes to that.

##### [00:26:27.060] - Maha Elkheir

Yeah, actually, to add a bit about the bottom up approach. I think it's also great, for instance, to have a sort of digital approach. For instance, we know that there is a Code.org YouTube page. So I think it would be a great opportunity, for instance, to have teachers use social platforms such as YouTube. Do you think it can be a great idea to mix like digital and life teaching when it comes to computing classes?

##### [00:26:51.240] - Ken Akiha

Yeah, I think, yeah, if I have two thoughts there, one goes back to kind of the previous question. But yeah, so I think definitely I mean, I think that's something that every teacher has had to learn over the past year, year and a half with the pandemic is is kind of if they weren't already incorporating kind of digital elements they're teaching, maybe they have had to learn how to do that. It just teaching on video calls, teaching, remotely teaching and these kinds of different settings and using the tools to do that.

##### [00:27:17.760] - Ken Akiha

And I think so. I think teachers have gained a lot of skills there and I've seen what works and what doesn't. In terms of incorporating digital elements into their into their teaching. And I think one example is just going back to like the how to leverage it, I think. I mentioned the students, but another thought I had, and I think something that you that I've seen some districts do in the U.S. is, you know, engaging parents.

##### [00:27:39.270] - Ken Akiha

And I think engaging parents through that can be sometimes done better digitally than it can be through in-person writing, just scheduling and whatnot. It's sometimes hard to get parents in the building or get them face to face to talk with them. But engaging them in whatever way that might be, it might be through something like, you know, a Code Week activity or maybe something more, just kind of like a parent night, something like that. But hearing from your, from the community of parents in your area is something that I've found to also be an effective way to to get leverage in saying like, this is something that we want.

##### [00:28:13.170] - Ken Akiha

And, you know, the YouTube page, I think I talked about the videos earlier, but that's something that I've always, you know, both as a teacher and then now as somebody working on the curriculum. You know, I think our videos are something that is really a big asset that we have as an organization. We have a team of folks who make the videos and they're really well done, and I think the videos can serve a lot of purposes.

##### [00:28:35.290] - Ken Akiha

So like I mentioned is we're really trying to support teachers who might not have any background in computer science. And so one one use of the videos is definitely to help support in terms of content delivery. Right. If if you don't, if a teacher maybe doesn't feel super confident about explaining the details of the I keep coming back to this, but how the internet works, right? We have a whole series of videos that explains that and those are those are there on the YouTube page and they can be used, however, but they're also then embedded in our lessons so that teachers feel like they can focus more on the activity with their students engaging their students.

##### [00:29:09.690] - Ken Akiha

And then if students might miss some of the bits of content, those videos can kind of fill in any of those gaps that might have happened throughout the throughout the lesson. And so, yeah, we design our videos with with that in mind, with trying to help support that implementation and support and let teachers almost give them permission to not be like the sole expert on everything and let the video take care of some of it because we know we'd rather have them focus on being present with their students, engaging with their students and maybe the activity that came before that, that's going to be a lot better use of their time.

##### [00:29:45.540] - Eugenia Casariego

To sum up our conversation, I would say that we had, what we have discussed today and what we I think I think is the main takeaway messages from this conversation is that not only how many ways are there for us to integrate coding and computer science in general into teaching. So we have discussed, for example, project based learning we have discussed as well playful learning. But as well, I think what I take from our conversation today with you, Ken, is that there's already a lot of resources and there's already a lot of support to teachers and that's already there.

##### [00:30:15.990] - Eugenia Casariego

And then we are going halfway there, while understanding, of course, the obstacles that students and teachers may face. And I think this is something important that we have discussed today that there are obstacles and that we acknowledge them and that we are actively working on that. But Maha, what's your, I don't know, take away message from today's conversation.

##### [00:30:34.710] - Maha Elkheir

So to add up to what you have said, I think that for me, the main two points that we have today is, of course, there is a policy level needs to be worked on, but also an important human level to this. So not only we have the teacher will also have the students. It's pretty important to work on the accessibility as well as the sense of community. And I think that's common, whether it is the US or whether it is in Europe.

##### [00:30:55.590] - Maha Elkheir

I think it's a worldwide issue, as we can say.

##### [00:30:59.100] - Eugenia Casariego

Indeed, indeed. And Ken, would you like to add anything else before we close the episode?

##### [00:31:03.630] - Ken Akiha

I think I would just say that I really like that point that you made, but like the human level, and I really think that in the end, that's that's a lot of what motivates the work that I do. And and I think what I would just encourage people to try to connect, right? Yeah, it's great to teach teaching CS or doing a code week activity. All of these things are great. And I think in the end, the reason that they're great is because they allow.

##### [00:31:28.800] - Ken Akiha

They allow us to connect with with students in a different way than we maybe have before.

##### [00:31:32.700] - Eugenia Casariego

Thank you. Thank you so much for joining us today. Ken it's it's been a pleasure. It's been great to hear more of the work that you do at Code.org, and that your impressions on the current status quo as well in the future as they took place as a thank you very much. So we are now coming to the end of the episode. We hope that you have enjoyed it and that you have been inspired to dive deeper into the exciting world of coding.

##### [00:31:52.920] - Eugenia Casariego

And as well they have gotten a bit more familiar with the important work that Code.org is doing in the U.S., but as well internationally. Don't forget that you still can access freely the resources of Code.org, as well as, of course, Code Week. So we invite you to check out both websites, both resources and to use them on your teaching practice.

##### [00:32:10.350] - Maha Elkheir

And we hope also to see you next time for a new episode of Code Week podcast with some interesting facts on coding and digital technologies in education. Goodbye.

##### [00:32:19.290] - Eugenia Casariego

Goodbye.