# Transcript: IT Podcast - Ep 70 - C951 Intro to AI Deep Dive - DeNece and Jim

*The following transcript is a verbatim account of the video or audio file accompanying this transcript.*

Speaker #1 (Narrator):

WG use IT audio series. Flexible, portable, profound.

Speaker #2 (DeNece Meyer):

Hello, This is DeNece Meyer and today I want to talk to you a little bit more. I know we have a podcast on just the overall Intro to Artificial Intelligence, C951. But today we wanted to do a little bit Deeper Dive intothe individual task and some questions that students may have in the podcasts. I do have Jim Ashe here with me today, a course instructor in the BSCS program and he's going tohelp me with some questions on these tasks. Jim are you there?

Speaker #3 (Jim Ashe):

Yes, I am. Thanks for having me again, DeNece.

Speaker #2 (DeNece Meyer):

Well, we have a few questions that students have come up with with the individual tasks. First thing I want to ask you right off the bat, we'll start talking about Task 1. But what is some general advice about Task 1 and Task 2, maybe even Task 3 that you can start us off with just togive us some good advice on starting these tasks.

Speaker #3 (Jim Ashe):

Oh yeah, that's a great question because there's often a little confusion about, you have the task directions and then the rubric andthen this evaluation process and how that happens. Just some general advice about dealing with the evaluations, the rubrics is what they're supposed to actually grade by. These are checklists. Most of these items you look at Task 1, Task 2 to a lesser extent Task 3, although quality there is assessed. But for Task 1 and Task 2 especially, they are checklists, they're not very specific details abouthow complex things need to be, they just need to be there. Quality generally isn't accessed. They don't say how fancy your rescue robot needs to be, how complex your chatbot needs to be, only particular things that it needs to do. If it does that, those items and the rubric will be checked off. Now, on the other side of that, you cannot pass the Turing test for the chatbot and outsmart Skynet or whatever. You can invent AI but it only recommends four jobs instead of five, like they specify, it won't pass. So everything matters. They count these, there some of them. When you go through the rubric make sure, hey, I did this, if there's anything that's specified, try to meet those things specifically. For each of those please try to do the minimum. Whatever you think is a minimum, try to at least do the minimum and I think is somewherethe first emissions are easier to pass. You might imagine, hey, I've got 10 items here that they have to check and you did nine, just fine and then the 10th is a little gray area. You might imagine them I give you a little bit of a pass on that when itgets sent back and there's an item that's marked. If it's in red, well they're only going to look at that. The stuff that's in green is done. You don't have to do it. But if something's marked as yellow or red, that's the only thing you need to address, but the evaluator on the second or third attempt or whatever, will only look at those parts that are marked. In other scope is more focused on that. Then on the other hand, don't overkill things either. This is a common mistake students make with all these taskis going way over the top with stuff. If you think you've met the minimum of the requirements, great. Now, submit it. It would be easier, I think in most cases to do what you think the minimum is, send it out there and then let them send it back and say, hey, these two or three things need to be addressed, updated a little bit rather than overkill the entire project.

Speaker #2 (DeNece Meyer):

That's great advice. It brings me to the thought of my students when I'm talking to them on the phone, they're always saying, I'm not sure when to reach out to that course instructor. I'm not sure really when to connect. I don't know if I know enough to ask the questions yet. How do I address that with them on exactly knowing when it istime to reach out to you on any one of these?

Speaker #3 (Jim Ashe):

Well, of course, 'Hey, here's a rubric item. I'm not sure what that means. Reach out email or appointment, ' that be a great thing to ask about. But often, I think some of the best conversations, most helpful conversations we have is conversations, I really don't know what I'm supposed to view here. In order to do these tasks correctly, you need to know what problem you're trying to solve. You need to know where that bar is, what it is you're actually looking for. If you're confused about that, yes reach out. It's what we're here for. It is literally our job to talk aboutthese projects and help these students get started on it. Make it a 30 minute or 45 minute appointment. But often it's just a 15 minute talk, for straightening out the direction that theyneed to go and giving them confidence enough to get started. Once you get started, the ball gets rolling, things usually happen and then start taking care of themselves. That's a great time. I think the more you are confused about the question, sometimes you're like, well I don't have a question, I just need help. That's a great time that would reach out to a course instructor.

Speaker #2 (DeNece Meyer):

Great idea. I know in this Task 1, a lot of students will ask me questions about the bot. What about the bot? What do they need to know?

Speaker #3 (Jim Ashe):

Well, it needs to work. For this Task 1, one thing you want to do is avoid making it an AI ML course. That's the scripting language that are used to write the bots behavior. You want to look at what the requirements are, you need to recommend one out of five tech jobs. They're not going to assess whether this is great career advice or not. That's not what this project is about. It's about having a functioning bot. It's good to know that the bot itself almost always passes. They don't say how complex your advice needs to be. They just need to see it walks a student through, gives him the career advice. You have to have a way to come toeach job conclusion and then pick one out of those five tech jobs. That part usually almost always pass. It's usually some detail with a document that gets sent back. I do advise students, well first meet that minimum. I think after that, you can make the bot is fancy as you like. With all these projects, all our WG projects, you can make them as fancy as you like. You can keep going off into space and get really, really complex, and that might be a nice portfolio piece, but start with making sure I meetthe minimum requirements so that I'll pass and meet my personal goals with this course.

Speaker #2 (DeNece Meyer):

Great points. I know students always ask me all the time, and I'm always looking through the course itself, but they're always asking me, is there any additional videos or resources for this task?

Speaker #3 (Jim Ashe):

For Task 1, yeah, I think we have really nice couple of videos. One we have this intro video talking about what those minimal requirements are, given you some examples of what they'll look like goingthrough the resource that are used itself. It's a Pandora site where you construct this. You don't need to deploy the bot. That means put it on a website where everyone can do it. It goes directly through the, well, it can't be deployed and it is deployed now directly on the Pandora site itself. The bots or chatbots in general typically take in training data. People use them and then they learn how to better do whatever they're trying to do. We can't do that here because we don't havetime to put our bot out there and have a bunch of people use it. We don't have a big trove of data to work with. We emulate that by getting it set up and use some specific examples with it. The videos go through that and they explain some of that. We also have, and I think this has beenone of the biggest things to help with this course, is an alternative set of directions. You have the directions and a task directions, and the rubric, and there is some difference between those and it's often confusing so we've writtenan alternative set of directions with a nice little FAQ at the bottom. Within that document, most students get this as soon as they start the course. If not reach out to your course instructor and get a link, it's a Google doc. All these resources that I mentioned are linked within that document so it's a one-stop place. Get all those resources and you can also see some different, and I think more accessible explanation for each of the rubric items. It's a great place to start.

Speaker #2 (DeNece Meyer):

Good point, I love those tips, Jim, I really like to be able to share those with students. Let's talk a little bit about task 2. What are some good points about task 2?

Speaker #3 (Jim Ashe):

Task 1, don't make it an AMI course. This course uses a CoppeliaSim robot simulation program. It uses this Lua scripting language to write the logic of the robot, which you're going to do and you need to pick outa disaster scenario and a job for your robot to do. It could be rescuing people, finding the way out of a fire, putting out fires, it could be anything, that's a little open-ended. I think sometimes the mistake students make here is they start with the job, "Hey, I want to do this, " which is cool it's a good way to go. Usually, I'm going to start with this job that sounds cool, and then they get half intothe complexities of the program and that would be a course in of it by itself. I advise students to start with the tutorial files for this course. It comes packaged with working robots. You can look at those working robots and then you can look at their code and see howit works and then you can modify their code to do stuff that you want to do. I think we can all appreciate that as computer scientists we're good at learning. That's what a computer scientist should be, you should be a problem solver, you're not a Java person or a Python person, you should be used to picking up different skills and languages. But it's usually easier if you have a working example tostart with and modify working code. That's what this tutorial files allow you to do. Look at those tutorial files, say, "Okay, here's stuff that I could do and modify, " and then use that to pick out your disaster job, what it is your bot has to do, your robot. You can modify these tutorial files and then move them over to your bot, press the play button, and watch it work. It doesn't have a robust debugger, that's another issue with it. Tweaking an existing code is a way easier way to go.

Speaker #2 (DeNece Meyer):

Right. Do you have any great points on task 3? I know it's probably the newer task in there but do you have any great points for that?

Speaker #3 (Jim Ashe):

Well, first just going back to task 2 we alsohave an alternative set of directions for that, and that's the place to start with some nice intro videos. I'm giving advice about how to get started with the projectwhat it is and also get started with the coding side. But task 3, advice about that, and we talked about this in the last video. I think one of the big question people always have is, "Well, what's this about and how does this connect to my Capstone? "For this project, you need to write a proposalfor a machine learning solution to some business problem. You pick out the business problem, you figure out a machine learning solution, and then you write a proposal. This proposal is going to include things like scope and budget, that's all made up. Hours that programmers are going to spend, how much you're going to pay them, that's all made up. But the machine learning process should be something that theyrecognize and can be applied to some business problem. This is what you do for your Capstone, well here it's all theoretical, in your Capstone, you're actually going to have to have real data, you're going to have to have access to data and actually do the solution. Here we just write the proposal.

Speaker #2 (DeNece Meyer):

Great. What about things like maybe the student gets sent back for articulation?

Speaker #3 (Jim Ashe):

Well, usually it's for grammar. Grammar or sources cited for both task 1, 2, and 3 is the most common thing I see sent back. I know it's easy to get involved with the heavy lifting of the project, it's easy to forget about the grammar. Use grammarly. com, what the evaluators use. I don't know what the threshold is but like two or three things remarked, they'll automatically send it back no matter howgood the rest of the project is so use grammarly. com. The free side has been more than sufficient. Sources get sent back. But often students will say, "Hey, I used this as a guide, I'll list it as a source", but it doesn't have a matching in-text citation, that'll automatically get sent back. If you don't have a matching in-text citation for your source, that'll get marked, if it doesn't deserve an in-text citation leave it off. If it's something that you use for your code, put it in as a code comment within the code itself, that would be the way to do it. That's the minor things that I see most common but beyond that, this task is a little different than the others. Well, the others are more of a checklist and if it meets it, they say this and they see then they mark it off usually and if it's done correctly. Here they do commonly see things sent back for insufficient details. That just means they want more. The number of words seems to matter here, and this is true for the Capstone as well. In that case, well just write more. Again, if you think you've met the requirement it's enough, do that. If they're sent back, well, we need more details on this, well just give them more details on that one section where they're asking for details.

Speaker #2 (DeNece Meyer):

Great points, Jim and I thank you so much for your time today. I think that wraps up our time that we have sodefinitely want to go forward in some other ones. Do you have anything else you want to add to that Jim?

Speaker #3 (Jim Ashe):

Just real quick about the evaluation. When they get sent back some of the comments thatthe evaluators write are a little confusing or vague sometimes. Keep in mind that they're following a template they're required to. They're required to acknowledge something that you wrote and thenthey're supposed to say something that is missing or wrong, in the case where they wanted more where they write insufficient details. I think that's a little confusing sometimesbecause they're like, "Well, acknowledge this", and it sounds like they're saying that it's correct or required, that may not necessarily be the case. You've written a lot, I don't know, like a history of Mickey Mouse for your stuff, supposed to talk about optimization. They would write back, "Well, the student has a very nice history in Mickey Mouse, but the items about optimization are missing. "That's like a required format that they follow. They try to be objective as possible, of course, that's very difficult especially on task 3, where there's a lot of gray area, but they do try to be as objective as possible. When you submit it for task 1 and then maybe it got rejected and then you submitted again, it will probably be a different evaluator, it's a ticket system, they just take the next one in line. You won't necessarily be dealing with the same evaluator repeatedly. It doesn't work that way for them you're not assigned a particular evaluator like you are a course instructor.

Speaker #2 (DeNece Meyer):

Well, thank you, Jim that's some great information. Thank you for joining me today. Is there anything else that we need to add to it? I think you've answered a lot of my student's questions.

Speaker #3 (Jim Ashe):

Just get started.

Speaker #2 (DeNece Meyer):

Exactly.

Speaker #3 (Jim Ashe):

A lot of times they get stuck up front and it's like, "I'm not quite sure what I need to do. "In that case, hey, talk to us. We'll try to limit the scope of what you're looking at and just get started. Once you start writing that bot, it takes care of itself. Once you work with the tutorial files and get things rolling started with the robot, it starts to take care of itself. If you're having trouble with task 3 because a lot of that is just making things up. We're not used to a creative writing assignments, just start writing one section at a time. Align it with the rubric, this is a great way to do all these tasks document, align them with the rubric. Use a rubric as section headers and then write that section, then take a break. Then write the next section. Come back the next day. If they send it back and you've got it aligned like that, part C they've marked as yellow you'll know right where to go inyour document and then update it and make some changes.

Speaker #2 (DeNece Meyer):

Great points, Jim. Thank you so much for joining me today. I know this is great information forour students and I can't wait to get some more feedback, so look forward to talking to you soon.

Speaker #3 (Jim Ashe):

You too DeNece.

Speaker #1 (Narrator):

Schedule time with your course instructor to explore more deeply. WGU, a new kind of you.