Person1: From your point of view, what is the sustainability in terms of software?

Person2: It’s -- as a sort of working developer, it's not something that I have to consider that much. Um, I appreciate that from a business point of view it's a very important uh, you know, consideration but it's not one that I've had to face um, pretty much at all.

Person1: Hm.

Person2: Um, I guess it would involve things like, um, do you have any particular operating system requirements? Do your -- you know, does your software depend on a particular uh, version of an operating system? That sort of thing. Um, or whether the services that you rely on, you know, whether they will be always available. Um, so as I say, it is not something that I've had much experience with. I see it more as a, a management function rather than a technical function.

Person1: I see. And what are the attributes or features of the software itself that led you to believe that it is sustainable?

Person2: Could you repeat that one, please?

Person1: Um, so if we -- we have a piece of code, of software.

Person2: Yeah.

Person1: And uh, is there any attributes or features of that software that led you to believe that it's sustainable?

Person2: Um, that make me believe that it is sustainable?

Person1: Yeah.

Person2: Well--

Person1: For example, documentation?

Person2: Yeah, um, the first thing you'd have to sort of analyse, are what are it's dependencies, if you like? Um, and that can be at different levels not eh -- but if it's a self contained module, what operating system does it run on? Does it require any special hardware? Um, how long lived is that hardware or operating system likely to be? Um, I can't think of anything much else.

Person1: That's all right. Regarding the software that you developed, then, sustainability, was sustainability a consideration? I guess not, right?

Person2: No, not for me, uh, this is a completely new stack, the mean stack. It's the first time I've used it. So the focus has been much more on the excitement of getting to develop with a new stack, um. I haven't consciously thought about sustainability. Um, my colleague, ***, who is my manager, he may have thought about it. I think with a lot of software projects, we go along and we just pick the, a technology and then become much more focused on getting something that works, that does what the user wants rather than, think about you know, the sustainability aspects.

Person2: Thinking about it quickly I would say that because part of the stack is within the browser here, because its a web application then, you would hope that the other developers would keep you know, backwards compatibility. Um, a slightly different question for us is what level of browser do you want to support? How far back do you want to go? That's sort of backwards, you know, sustainability if you like. Um, and that obviously add a layer of word and you complexity for us. Because you know, the further back you need to go then the more gotchas there you know, for the web developer.

Person1: Mm-hmm (affirmative). And have you worked on any projects that were not sustainable?

Person2: Um, no. I've never really run into the situation where we've come up against um, real issues. I'm going to think. There was one case many years ago where I was working on a project for Brand3 and Brand4, where we had to take over basically the project that moved from one country to another, and we had to take over a piece of software. And that piece of software had been written using a certain technology that was aging at that time. So that would be my only real direct experience. We had to make a decision as to whether to support that software as it was or whether we needed to migrate it to a new technology.

Person1: Mm-hmm (affirmative).

Person2: We had to, the decision was, was to maintain it as it was in the first instance, but a second team was tasked with starting to migrate it so there was like a parallel work stream which was I suppose quite expensive then, in terms of labor and you know, developer time. So in a new sense, that was a sustainability issue because it was recognised that maybe there were two business cases here. One was that we need to use the software as is and to make sure that we understood it and that we could support it as it was. But it would also seem that this was not valuable in the long term, that it needed to be migrated.

Person1: I see. Were there any consequences of it not being sustainable?

Person2: Yeah, the consequence there was that there was an increased burden and cost to the organisation. We have to invest time and effort in learning the existing product, just to make sure that we can get things done in a timely manner, but as I said, there was a second team that was tasked with looking into how we could replace this product or, you know, component of the product. So I guess it would be cost in the long term.

Person1: Mm-hmm (affirmative).

Person2: The other issue I could think of maybe where, I remember working for Company1, where we, one of the um, companies, which were part of uh, the Company1, um, organisation, we had to take over one of their products. I remember spending a couple of months writing documents, analysing every single module and itemising, just basically, doing reverse documentation, it was engineer, you know, documentation, just to demonstrate to the management whether this software was usable or not. Eventually a decision was made whether to try and migrate the existing product or to start fresh. Um, and that uh, I remember back, the decision was in the end to start afresh.

Person1: I see. Well, that's all I have for you today.