Subject: URGENT: PoC Needed for Trade Show

Hey,

As you know, I’m traveling between trade shows and I need a small PoC ASAP for the next one. We’re showing off the next line of networking devices (routers, switches, cameras, door access systems), and I need something that makes our hardware look smart.

We need a device monitoring service that checks if a list of network devices is alive. It needs to support both gRPC and REST calls, retrieve the diagnostics data (HW, SW, FW version, status, checksum) depending on protocol the device supports – use its health endpoint to get the capabilities.

It should be simple but look professional since this might become part of something bigger. We need an API to retrieve the latest status of all monitored devices. The device list will change, so it should be easy to update dynamically without restarting.

It needs to be *fantastic* - remember potential customers will judge us on it! If device is down, we need to handle it intelligently—maybe retries? Maybe logging? Not sure, you decide - I don't have time to figure it out. Keep in mind that some devices are behind unstable networks, and we don’t want false alarms. Keep the device identities, status, etc stored in the Postgres database so that we can inspect them offline.

You must also integrate the server library for the devices with an external checksum generator binary executable. I heard it’s difficult to configure and setup, but I don't have it yet, so just focus on preparing the full PoC. Important note: It must be paired with the PoC life-cycle to get valid test results. Testing the whole thing must be easy! We’ve already received some of our gear here at the convention center and I anticipate a bunch of different devices and operating systems for hosting of the service.

I know the time zone gap makes this tricky, but I trust you’ll make something great. Thanks for pulling this together!

–Boss

Needless to say, you need to implement the go backend and provide a short reply to your boss. Try and immerse yourself into the situation and act as you would in a real-life scenario to the extent possible. It is fully understood that you have limited time and must prioritize – just as you would in a real-life situation.

Besides the technical requirements given in the e-mail there is also other less explicit information hidden between the lines. Read it carefully! Consider the unknowns. Decide on what the requirements are and what things to produce. You will be judged not only on the software you provide, but also on how well you translate soft implications and assumptions. You need to think about what good software is, how this code will affect your developer colleagues, the actual result at the trade show, the structure of your testing, the future of the PoC, developer experience and the future of the company.

Good luck!

NOTE: This test is meant to evaluate your coding skills. In order to do that, we need to see what you’re capable of building yourself. It’s all right to use AI assistance, but it has to be made clear where and for what purpose you have chosen to do so.