Hackathon Developer Notes

This note is primarily targeted at Hackathon participants that will be “coding” during the event.

The Hackathon challenge will involve a central service/process, likely running on the laptop of one of the participants on your team. The purpose of the central process will be to send queries to a number of devices in an effort to solve a problem. In this case, think of “device” as an abstract term. Many teams will be utilizing Raspberry Pis, and a Raspberry Pi would be a device. For teams not using Raspberry Pis, a laptop can act as a device. And, some teams will have a combination of Raspberry Pis and Laptops acting as devices. The assumption here is that there will need to be some code that runs on each device to respond to queries from the central service/process. A team will likely be writing some code that will be the central service – and other code that will run on devices.

Due to networking challenges that exist at many of the sites, combined with the fact that many teams will have participants and devices in multiple locations, a mechanism is being provided to facilitate easy connectivity between the central service/process and all of the devices on a team – wherever they may be. This mechanism is Azure IoT Hub. Please read the provided AzureIOT document for more details. Some template/example code is being provided for C#, NodeJS, and Java. And, Azure IoT Hubs and Device IDs are being created in advance and will be provided to all teams before the morning of the Hackathon.

Other things to think about:

1. The Pi Image for the hackathon includes
   1. Java and Maven
   2. .NET Core 2.0 runtime (runtime only, so coding must be done on laptop. Laptop needs .NET Core 2 SDK)
   3. NodeJS and NPM
   4. Git
   5. Python (however, no Azure IOT HUB example files are available)
2. The OS on the Pi is Linux. If you are rusty, might brush up on the basics…

If the “coders” on your team can’t use one of the languages for which Azure IOT HUB templates have been provided (c#, Java, or NodeJS), but do have another language/runtime that you are comfortable with:

1. Using the Pis may be unviable for your team, as appropriate network/port connectivity to the Pi devices from your laptops may be a problem at your site. Will vary be site.
2. Your best path may be to simply not use Pis at all, and just use your laptop computers – assuming that all of the laptops needed to act as either devices or the central service/process can see each other on the network. This viability of this approach may vary from site to site.

Talk with your team, and think about your approach.

If you have technical problems, please reach out to [hackathon@kantar.com](mailto:hackathon@kantar.com), and we’ll try to help.