**8 Memo**

To: Professor Pisano

From: Ben Brewer, Ib Chand, Alex Prior, Allen Zou

Team: NoiseHub Team 8

Date: 11/17/2021

Subject: First Prototype Test Plan

1. **Required Materials**

**Hardware:**

* Raspberry Pi 4B
* Thermistor
* 12W USB C Power Adapter (RPI)
* 4.7㏀ Resistor (thermistor)
* Laptop (AWS observing, Pi SSH)
* Mobile Device (App observation)

**Software:**

* Python Scripts:
  + Read thermistor data
  + Send thermistor data to AWS IoT core
* AWS Cognito User Authentication
* AWS DynamoDB User Preferences Data

**2.0 Test Setup**

The Raspberry Pi will be turned on and connected to one team member’s mobile hotspot through a monitor and keyboard. Next, the team will ensure the Lidar and thermistor are properly wired. Then, AWS TimeStream will be pulled up on one member's laptop, and the mobile app running on Expo Go will be pulled up on another member's mobile device.

**3.0 Test Procedure**

1. Start Expo server for mobile application
2. Set up AWS IoT Core and Timestream data monitoring
3. Place thermistor on table to get room temperature reading
4. Observe initial temperature being sent to AWS Timestream
5. Wrap thermistor in hand to raise temperature
6. Observe change in readings being sent to AWS
7. Create account on mobile application and view data
8. Observe continuous thermistor readings locally on Raspberry Pi
9. Place lidar sensor on table
10. Stand near the lidar sensor and observe initial distance reading on AWS Timestream
11. Move away from the sensor and observe changes in readings sent to AWS TImestream

**4.0 Measurable Criteria**

* Pi locally captures thermistor data continuously (every second) for ten consecutive data points
* Pi locally captures Lidar data continuously (every second) for ten consecutive data points
* Pi successfully sends data once every second for ten seconds to IoT Core over MQTT
* IoT Core successfully forwards received data to Timestream
* Thermistor measurements respond to environment changes
* Lidar measurements respond to environment changes
* User can successfully create account in mobile application and sign in
* Skeleton app is populated with data from AWS for ten consecutive data points

**Results before thermistor wrapped in hand**

| Data set # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pi captures Lidar data |  |  |  |  |  |  |  |  |  |  |
| Pi captures thermistor data |  |  |  |  |  |  |  |  |  |  |
| AWS receives data from Pi |  |  |  |  |  |  |  |  |  |  |
| App is populated with data from AWS |  |  |  |  |  |  |  |  |  |  |

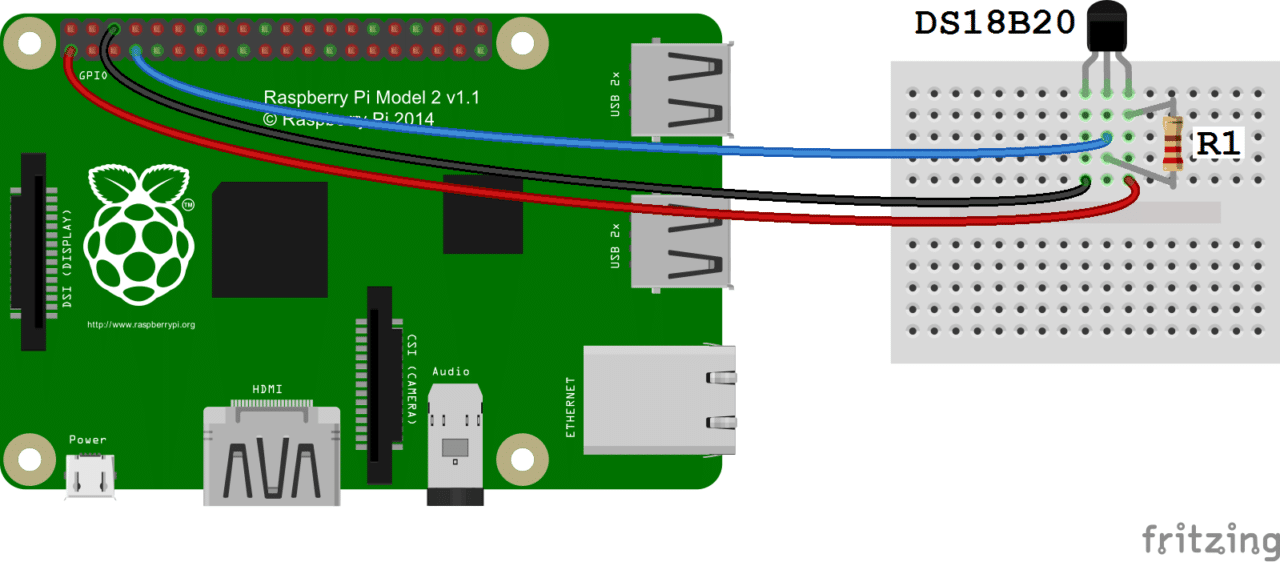
**Results after thermistor wrapped in hand**

| Data set # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pi captures changed thermistor data |  |  |  |  |  |  |  |  |  |  |
| AWS receives data from Pi |  |  |  |  |  |  |  |  |  |  |
| App is populated with data from AWS |  |  |  |  |  |  |  |  |  |  |

**Results after moving away from Lidar**

| Data set # | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pi captures changed Lidar data |  |  |  |  |  |  |  |  |  |  |
| AWS receives data from Pi |  |  |  |  |  |  |  |  |  |  |
| App is populated with data from AWS |  |  |  |  |  |  |  |  |  |  |

**5.0 Hardware Pinout**

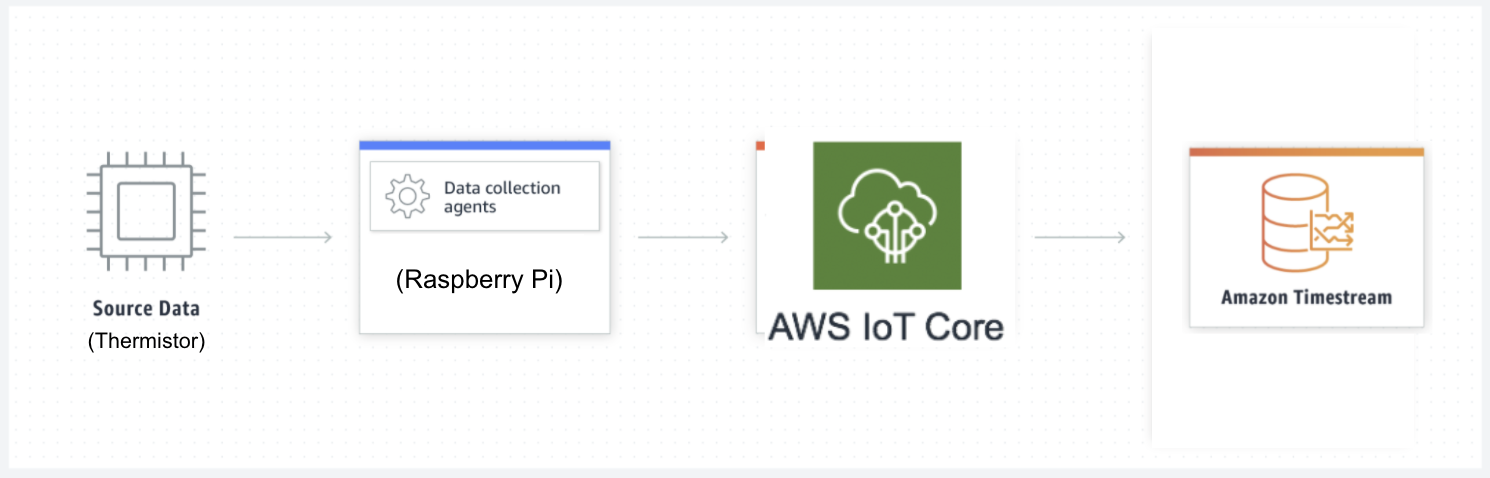


(Source: <https://www.circuitbasics.com/raspberry-pi-ds18b20-temperature-sensor-tutorial/>)

**Pinout Table**

| Raspberry Pi Pin # | Pin Description | Pin Usage |
| --- | --- | --- |
| 1 | Vcc | 3.3V Thermistor Power |
| 7 | GPIO #4 | Thermistor Data |
| 9 | Ground | Thermistor Ground |

**6.0 Dataflow**

****