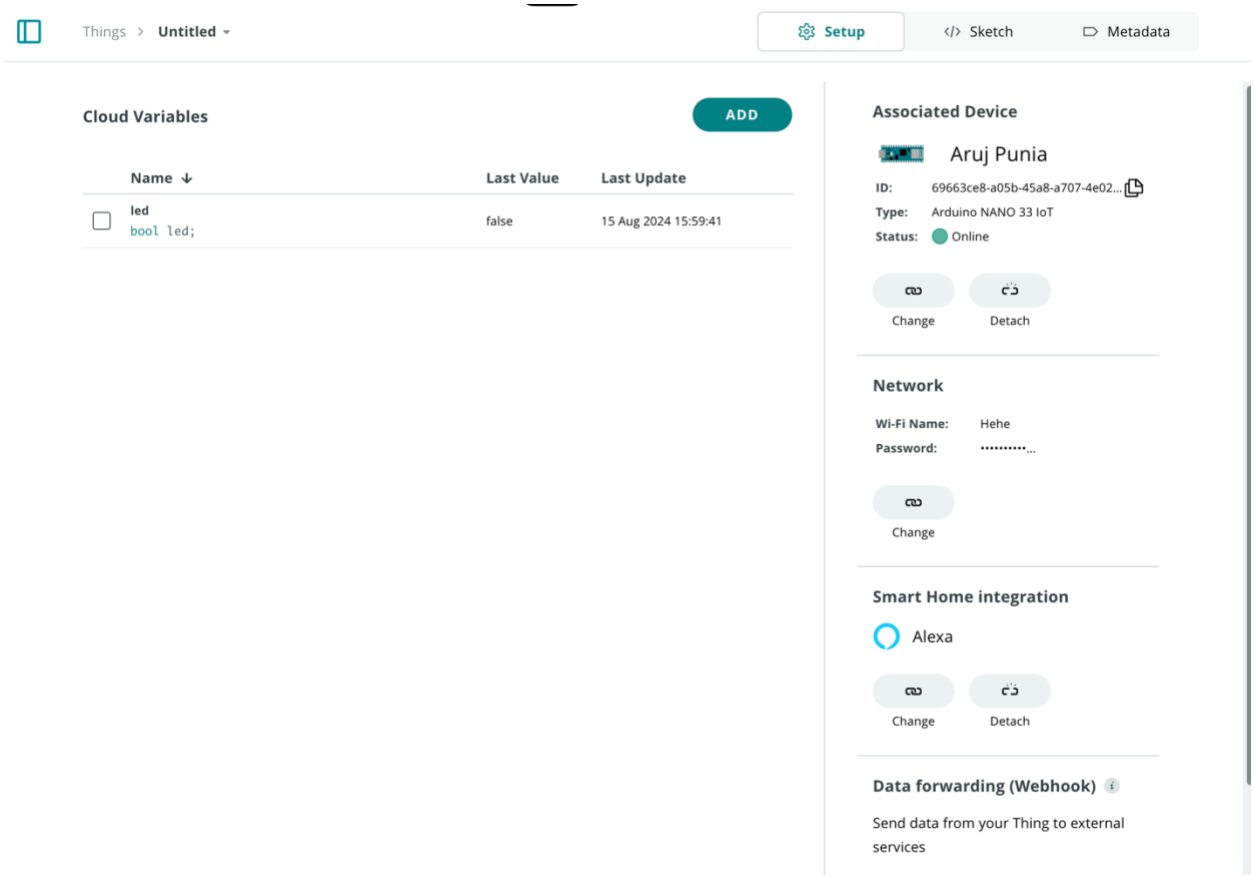
SIT 210 – Embedded Systems

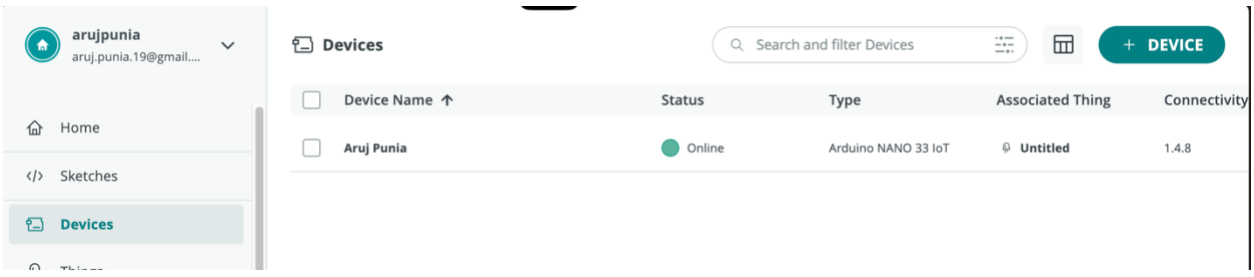
Aruj Punia

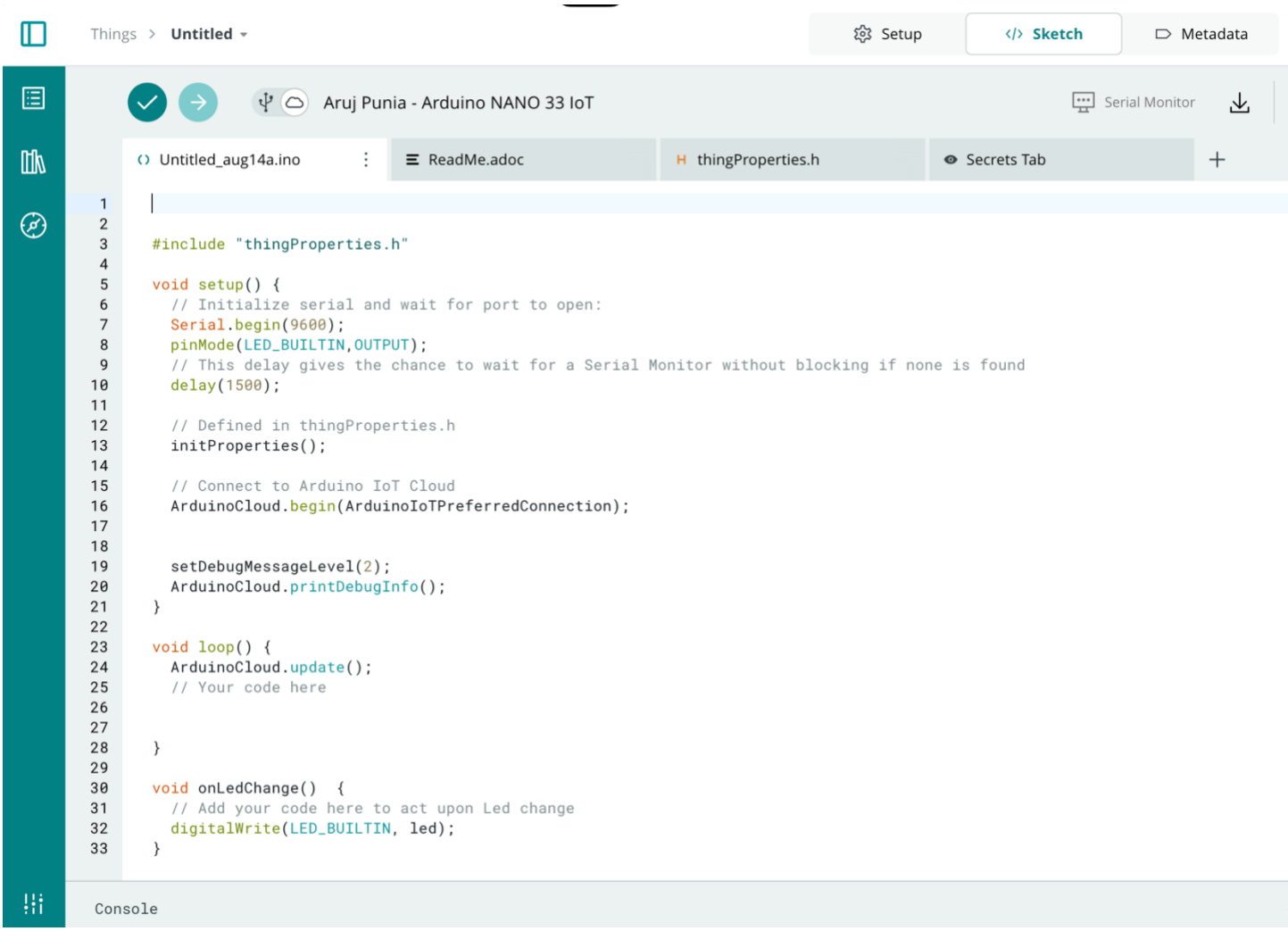
2310994767

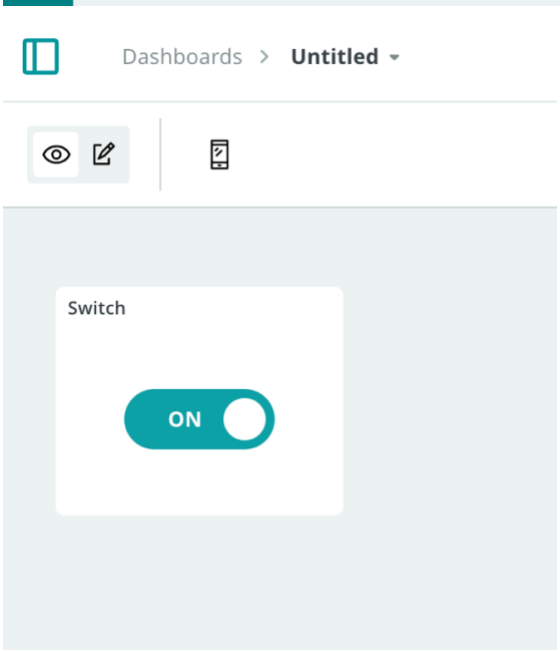
PART 1 –

Screenshots –






Things > Untitled
Setup
Sketch
Metadata
Aruj Punia - Arduino NANO 33 IoT
Serial Monitor
Untitled_aug14a.ino
ReadMe.adoc
thingProperties.h
Secrets Tab
#include "thingProperties.h"
void setup() {
// Initialize serial and wait for port to open:
Serial.begin(9600);
pinMode(LED_BUILTIN,OUTPUT);
// This delay gives the chance to wait for a Serial Monitor without blocking if none is found
delay(1500);
// Defined in thingProperties.h
initProperties();
// Connect to Arduino IoT Cloud
ArduinoCloud.begin(ArduinoIoTPreferredConnection);
setDebugMessageLevel(2);
ArduinoCloud.printDebugInfo();
}
void loop() {
ArduinoCloud.update();
// Your code here
}
void onLedChange() {
// Add your code here to act upon Led change
digitalWrite(LED_BUILTIN, led);
}
Console


Dashboards > Untitled
Switch
ON

PART 2 –

Describe how you would use modular programming to improve the usability of your Blink LED program-

1. To make the Blink LED program easier to use with modular programming, you can split the program into different parts, each responsible for a specific job.

1. Morse Code Module: You can create a Morse Code class that includes everything related to Morse code, like changing text into Morse code and timing for dots, dashes, and spaces. This module can keep track of Morse code patterns for various letters and offer ways to access them.

2. LED Control Module: Make a separate module to manage LED functions. This could have features for turning the LED on and off for certain lengths of time (for dots, dashes, and spaces). By separating LED control, the code becomes clearer and can be reused more easily.

3. Button Handling Module: Set up a module to take care of detecting button states and preventing errors from rapid presses. This helps manage button clicks reliably without making the main program too complicated.

4. **Main Program:** The setup() and loop() functions should mainly focus on working with these modules. The main program will start the Morse Code, LED Control, and Button Handler classes and use them to make the LED blink according to the Morse code when buttons are pressed.

This way of organizing the program makes it easier to read, maintain, and change specific parts when needed.

LINK FOR CODE-

LED BLINK MORSE CODE

LINK FOR VIDEO DEMONSTRATION –

YouTube Demonstration of Part 2