# IoT Methodology

Ann-Kareen Gedeus

February 2022

## 0.1 Overview

This project was designed to collect pulse sensor data from an Arduino MKR board and then send that data to the Blynk Cloud where that data would be grabbed and displayed on a Web app. Furthermore, the project specifies that users need to interact with the web app in order to change the state of the builtin LED on the device.

## 0.2 IoT Design Methodology

### 0.2.1 Purpose and Requirements Specification

**Purpose:** A pulse sensor system that monitors the BPM of users using a web application to display the data and control device led.
**Behavior:** The pulse sensor device should send it's data to the IoT cloud and that data needs to be displayed on the web app. Furthermore when the LED is toggled on the web app that information needs to be delivered to the cloud and the cloud will send that instruction to the Arduino device.
**System Management Requirement:** The system should provide BPM data and control of LED buttons.
**Data Analysis Requirement:** None.
**Application Development Requirement:** Application will live locally on device to display BPM data.
**Security Requirement:** Authentication token.

### 0.2.2 Process Specification

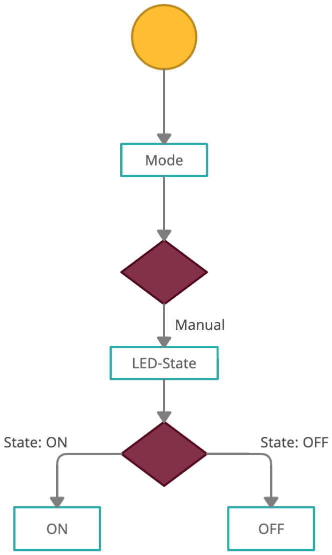


Figure 1: The figure above defines the state of the LED, known as the LED-State. The buttons on the web app and cloud define the led in two states: ON or OFF. The LED-State is manually changed by the user through the web app. If the user chooses the state ON the LED is turned on. If the user chooses the state OFF the LED is turned off.

### 0.2.3 Domain Model Specification

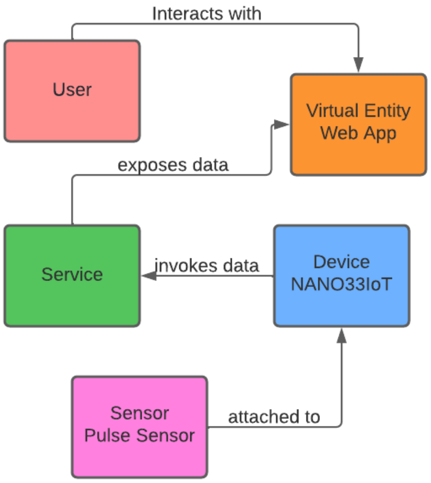


Figure 2: The figure above defines the attributes and relationships between entities and objects. The first relationship is between the user and the virtual entity. In this relationship, the user interacts with the web app either to view the pulse sensor data or to control the led. The second relationship is between the service (Blynk Cloud) and the web app, the service exposes the data (pulse sensor data) from its platform to the web app. The third relationship is between the cloud service and the device, this relationship allows for the service to invoke and get data from the physical device. The lat relationship is between the sensor and the device. In this relationship, the sensor is attached to the device and creates the conditions for data creation.

### 0.2.4 Information Model Specification

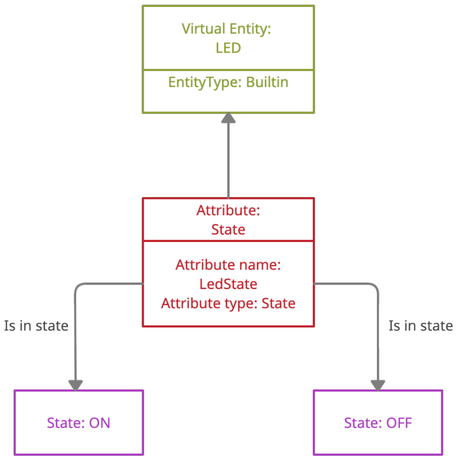


Figure 3: In the figure above the relations and attributes of the virtual entity are defined. The virtual entity defined is the LED and the attributed state given to the led is LEDState on the web app. This state has two components which are the ON or OFF state.

### 0.2.5 Service Specifications

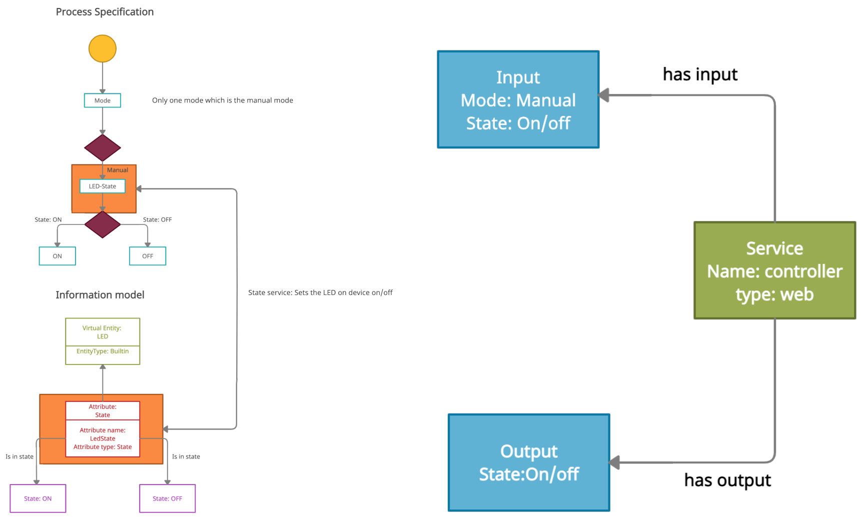


Figure 4: In the figure above the fifth step of the IoT design methodology is defined for our service specifications. The only service here is the led control on the web app. The service checks if the led has an input and output of ON or OFF.

### 0.2.6 IoT level Specification

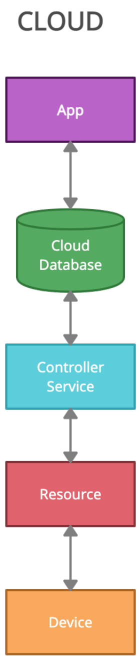


Figure 5: The IoT level is defined above. Our IoT level system only needs a cloud component. In this cloud level the app (web app) interacts with the cloud (Blynk Cloud) to send LEDState data and the cloud interacts with the app to send sensor data. This is connected to the controller service (Arduino IDE) which is connected to the resource (Internet, Memory, Processing) and ultimately the device.

### 0.2.7 Functional View Specification

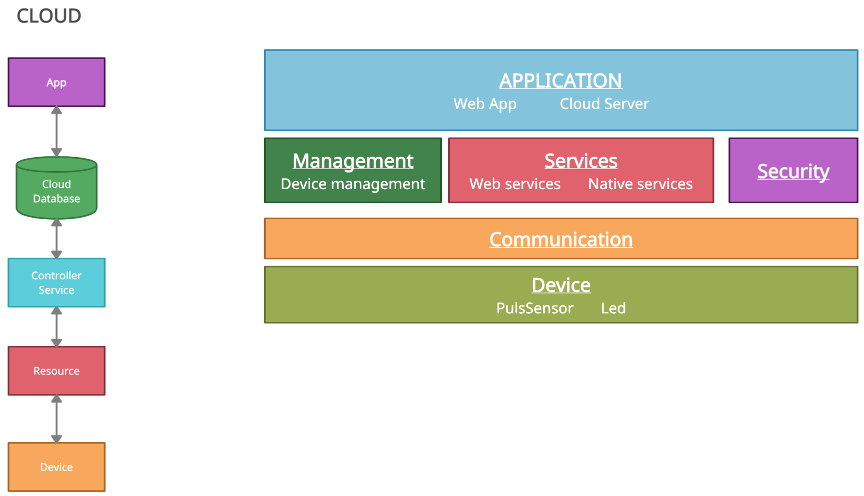


Figure 6: Defines the functions of the IoT systems grouped into various functional groups

- IoT Device maps to device functional group and to management functional group
- Resources maps to device and communication functional group
- Controller service maps to native services functional group and cloud database maps to web services functional group
- Cloud database maps to management functional group
- App maps to application, management, and security functional groups

### 0.2.8 Operational View Specification

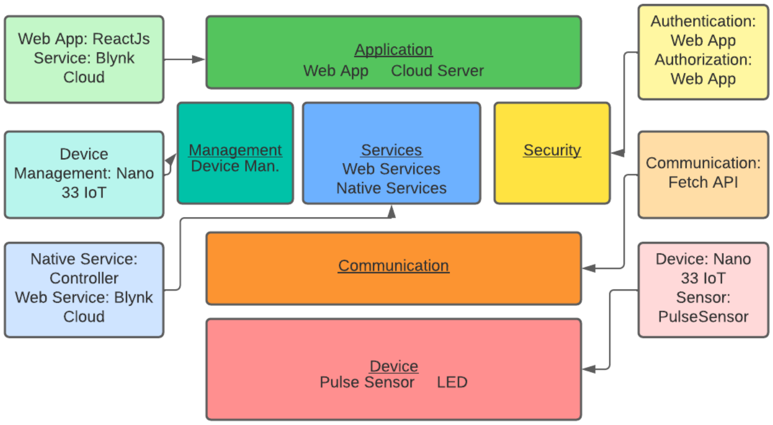


Figure 7: Elements of the IoT system are defined

### 0.2.9 Device Component Integration

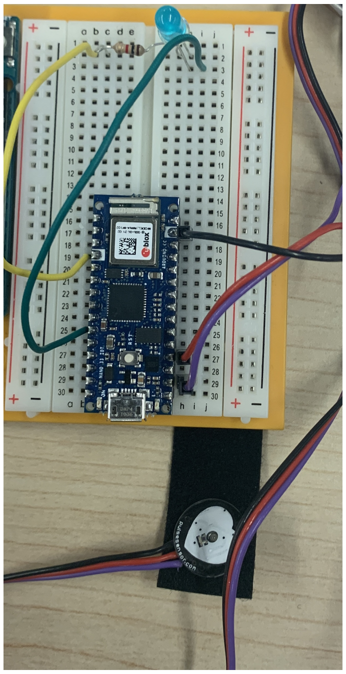
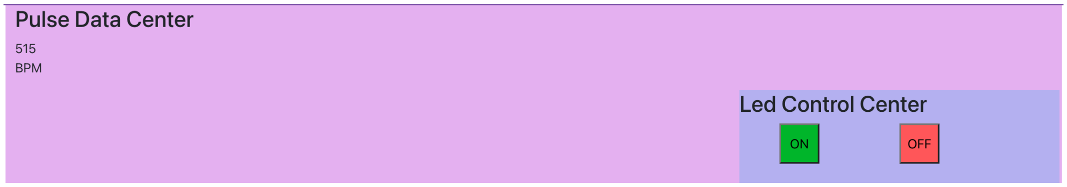


Figure 8: Device Component Integration

### 0.2.10 Application Development


Pulse Data Center

515
BPM

Led Control Center

Figure 9: Web Application using ReactJs and BootStrap

## 0.3 Schematic

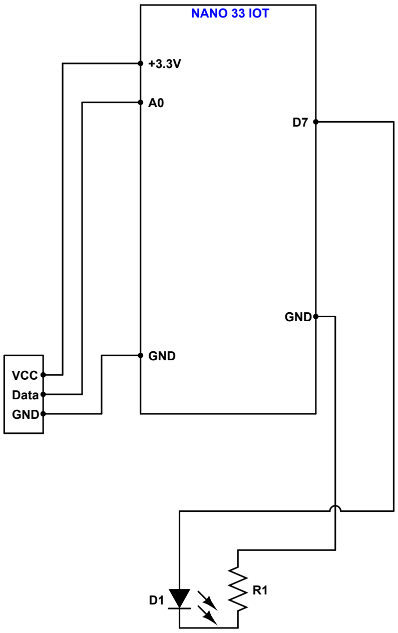


Figure 10: Circuit of Led and Pulse sensor

## 0.4 Bill of Materials

| Bom Level | Part Number | Part Name | Description | Quantity |
|---|---|---|---|---|
| 1 | 0-100 | Nano 33 Iot | Arduino Device | 1 |
| 1 | 0-101 | Pulse Sensor | Sensor | 1 |
| 1 | 0-102 | Arduino IDE | Software | 1 |
| 1 | 0-103 | Blynk Cloud | Software | 1 |
| 2 | 0-104 | Web App (ReactJs) | Software | 1 |
| 2 | 0-105 | Led | Hardware | 1 |
| 2 | 0-106 | Resistor | Hardware | 1 |
| 2 | 0-107 | Wire | Hardware | 2 |

## 0.5 References

1. Blynk API Documentation

2. LED Blynk Code

3. Blynk Sensor Code

4. REST API

5. Fetch API for beginners

6. Real-time data with Fetch API

7. Fetch API with ReactJs