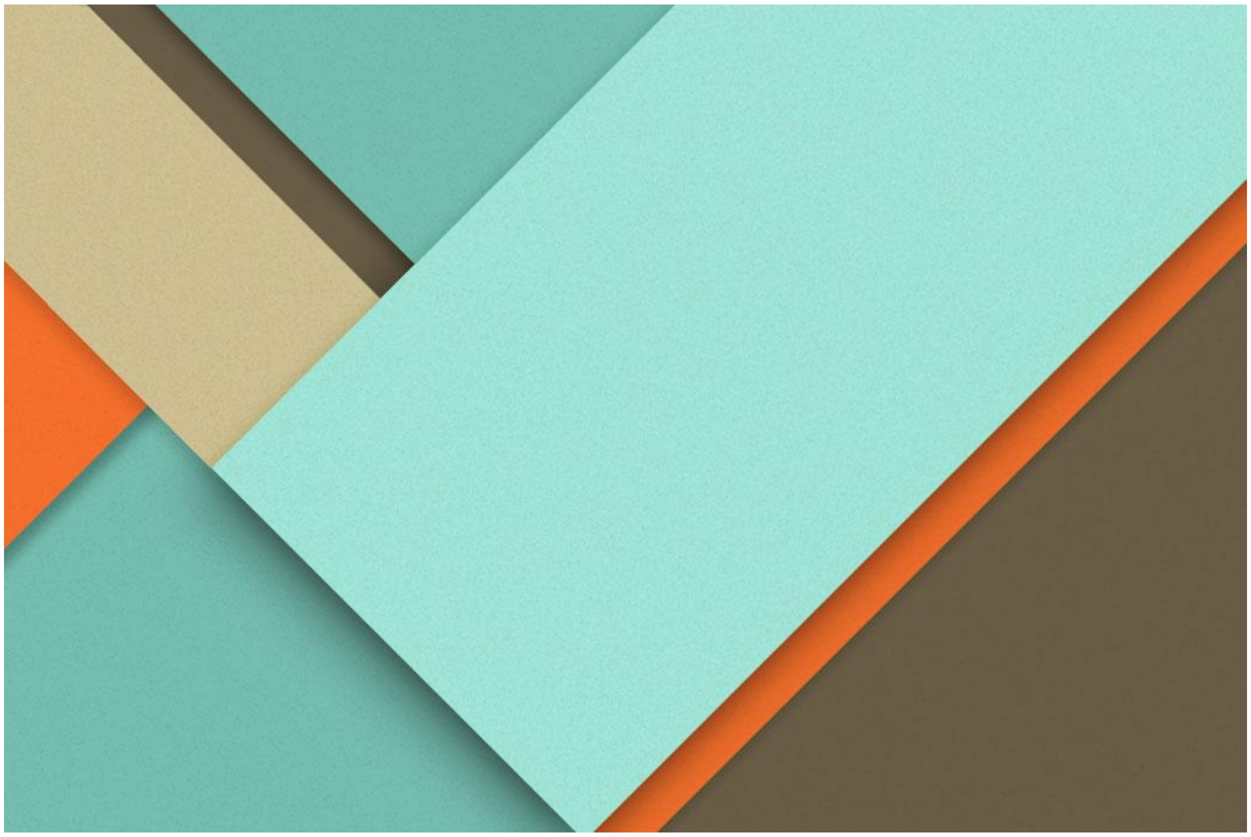


SmartSocket

Preliminary Report V1

26.04.2022

—

Nauman Shakir
https://NaumanShakir.com
https://3STechLabs.com

Overview

This preliminary report aims at defining an architecture for SmartSocket systems. The goals of this report are mentioned below.

Goals

1. Should have WiFi and eSim connectivity.
2. Should be small in size and can be used in place of normal sockets.
3. WebApp-based Client Portal, Admin Panel and a Smartphone app.
4. Each device should have a unique ID.
5. OTA functionality.

Specifications

The system is divided into 3 different layers, which should communicate with each other in real-time.

Layers

- Hardware - Sensor Nodes
- Processing Layer - Server {Ubuntu Server 20.04}
- Application Layer - smartphone app

Components Required(Prototype)

Using off-the-shelf components to test the first version of the device.

Microcontroller

- ESP32 PCI-E with SIM7600(no eSim)
  - https://de.aliexpress.com/item/4001142716386.html?gatewayAdapt=glo2deu
- Status LED Lights
  - https://www.amazon.com/MCIGICM-Circuit-Assorted-Science-Experiment/dp/B07PG84V17/ref=sr_1_3?crid=253CM8AP34PV7&keywords=led+lights+5mm&qid=1650112184&sprefix=led+lights+5mm%2Caps%2C199&sr=8-3

Power

- AC To DC Converter
  - https://www.amazon.com/Acxico-Converter-Supply-Module-Version/dp/B08545K26C/ref=sr_1_6?crid=31A7Q2G4SDHBG&keywords=220v+to+5v&qid=1651029482&sprefix=220v+to+5v%2Caps%2C156&sr=8-6

Relay

- SSR Relay
  - https://www.amazon.com/DollaTek-1-Channel-Level-Trigger-Module/dp/B07DK29FR6/ref=sr_1_12?crid=3AY7SEMFWEWVT&keywords=ssr+relay+module&qid=1651029377&sprefix=ss+relay+module%2Caps%2C174&sr=8-12

Components Required(Manufacturable Product)

Components for the end-product PCB(resistors, capacitors, inductors, connectors and other discrete components are not listed)

Microcontroller

- ESP32
  - https://www.google.com/search?q=esp32+chip&oq=esp32+chip&aqs=chrome..69i57j0i20i263i512j0i512l2j0i457i512j69i60l3.2759j0j4&sourceid=chrome&ie=UTF-8

Communication

- SIM7500E(eSim availability)
  - https://www.simcom.com/product/SIM7500X.html

eSim Test

Power

- AC To DC Converter 5V
  - https://www.amazon.com/Comidox-HLK-PM01-Household-Isolated-Step-Down/dp/B07LBM6PM7/ref=sr_1_6?crid=255XTJBN33PJE&keywords=220v+ac+to+5v+dc+module&qid=1651029860&sprefix=220v+ac+to+5v+dc+modul%2Caps%2C151&sr=8-6
- AC to DC Converter 3.3v
  - https://www.amazon.com/KOOBOOK-HLK-PM03-Intelligent-Household-Isolated/dp/B07TB61VC3/ref=sr_1_3?crid=20JR4OV2XA6DN&keywords=220v+ac+to+3.3v+dc+module&qid=1651029932&sprefix=220v+ac+to+3.3v+dc+module%2Caps%2C136&sr=8-3

Relay

- SSR Relay
  - https://www.amazon.com/Relay-Module-G3MB-202P-DC-AC-Solid/dp/B07K592TH7/ref=sr_1_21?crid=27FIJ3BDPDX2H&keywords=5v+ssr+relay&qid=1651029962&sprefix=5v+ssr+relay%2Caps%2C135&sr=8-21

Architecture

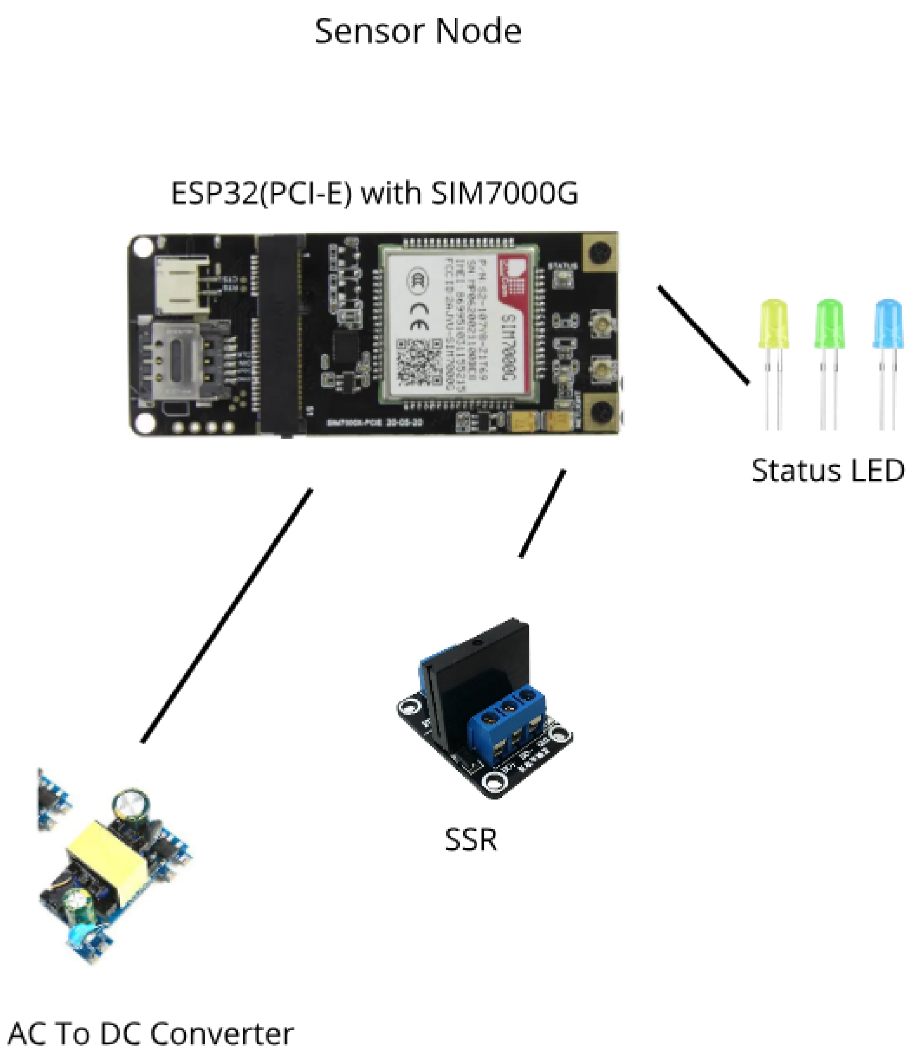
The complete project has multiple components

- Sensor Nodes
- IoT Server

Sensor Nodes is a completely enclosed SmartSocket device.

Sensor Nodes Architecture

(For Prototype)

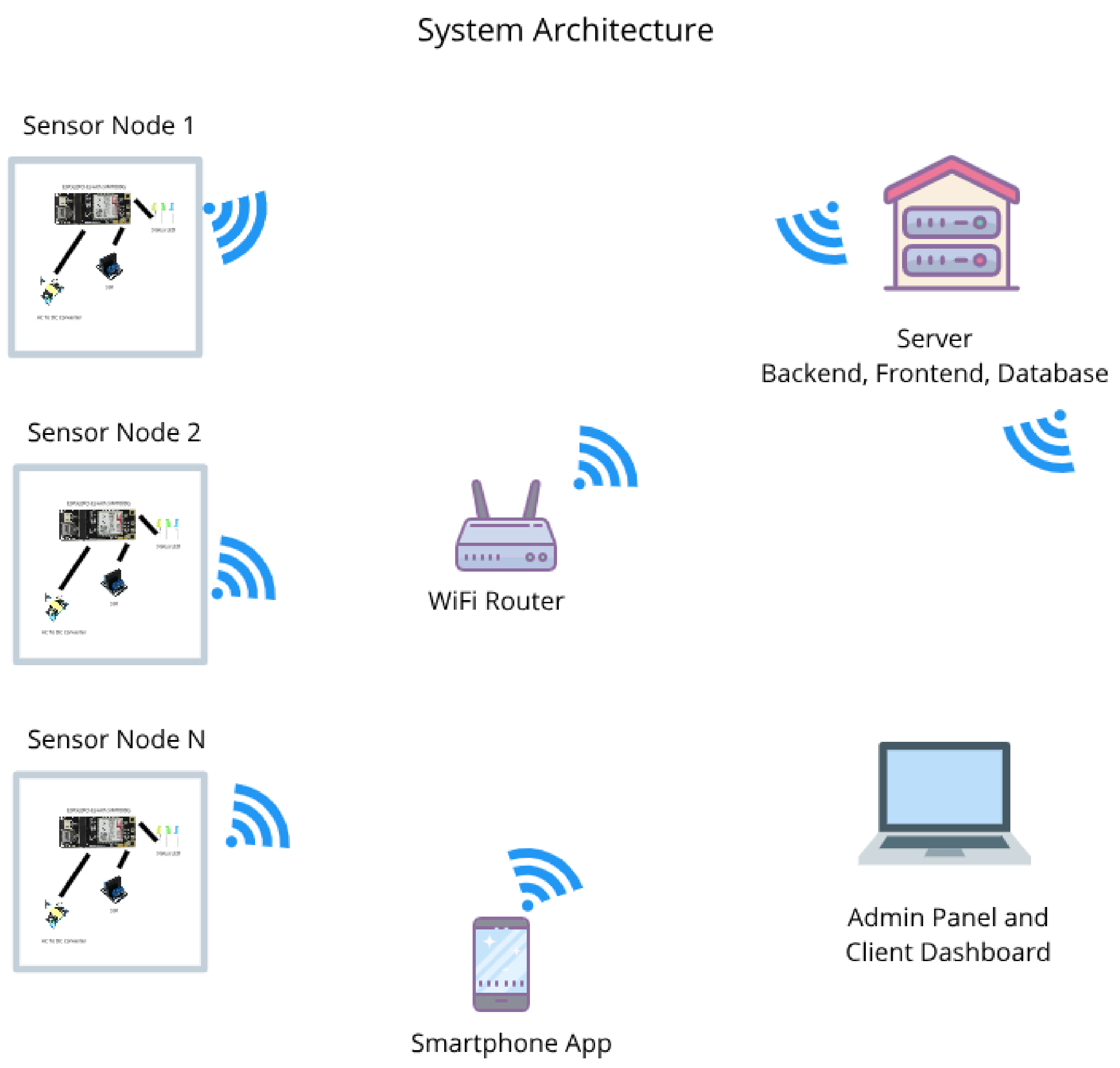


Above is the Sensor Nodes architecture. The sensor node is a single SmartSocket node. It contains ESP32 Board with SIM7000G for WiFi and 4G connectivity. Moreover, a Solid State Relay is connected to the ESP32 and the system is powered using an AC-DC converted module.

ESP32 is the MCU(Microcontroller) of Choice in this sensor node because:

- It has WiFi, Ethernet and BLE built-in.
- Has a multi-core processor and can handle data acquisition, data storage, configuration and communication with the server in parallel.
- Have plenty of GPIOs and can be used to add extra sensors in future.

System Architecture



Above shown is the complete system architecture. Sensor Nodes will use WiFi network/4G Connection for data communication with the Server. There could be multiple sensor nodes in the system. A web-app based dashboard hosted on a server will allow admins to see the devices from the database, while a client portal will be used by the clients to see the status of their smart socket devices and control them. A smartphone app will also allow clients to see the devices status and control them.

Software Architecture

Multiple Nodes management

The nodes should have unique identifiers, the MAC address numbers of ESP32, which will be used to uniquely identify each Node.

SensorNodeID;[DataString]

- SensorNodeID is a unique ID of Sensor Node
- [DataString] contains the sensor value(Status)

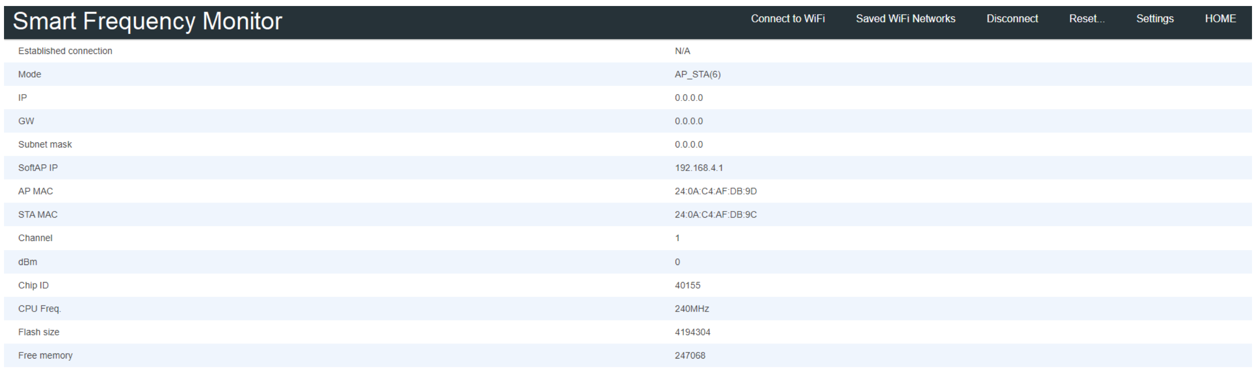
The communication is done over WiFi and data is shared to the IoT Server over the internet connection using the wifi network. The gateway will share the data with the server.

Captive Portal(Device Provisioning)

The sensor node can easily be configured. The flow will be

1. User power on a sensor node.
2. The sensor node becomes a WiFi Access Point.
3. The user connects to that AP and opens the address http://SmartSocket.local, showing a locally hosted webpage with all configuration options like connecting with a WiFi router, setting data acquisition frequency, server address configurations etc.
4. The user configures the sensor node on the captive portal and saves it.
5. The sensor node restarts automatically to connect to the configured WiFi router and hides its access point.
6. Users can now open the captive portal without connecting to the device access point because it is now connected to the WiFi router.

The captive portal will look something like shown below.



Established connection	N/A
Mode	AP_STA(6)
IP	0.0.0.0
GW	0.0.0.0
Subnet mask	0.0.0.0
SoftAP IP	192.168.4.1
AP MAC	24:0A:C4:AF:DB:9D
STA MAC	24:0A:C4:AF:DB:9C
Channel	1
dBm	0
Chip ID	40155
CPU Freq.	240MHz
Flash size	4194304
Free memory	247068

With a captive portal, users can

- Connect to new WiFi
- Connect to a saved WiFi Network
- Disconnect from WiFi
- Reset the Device
- Configure server connection, data frequency etc.
- Show real-time frequency data.

The captive portal functionality can also be directly integrated in the smartphone app for a smooth user experience.

Web Server Integration

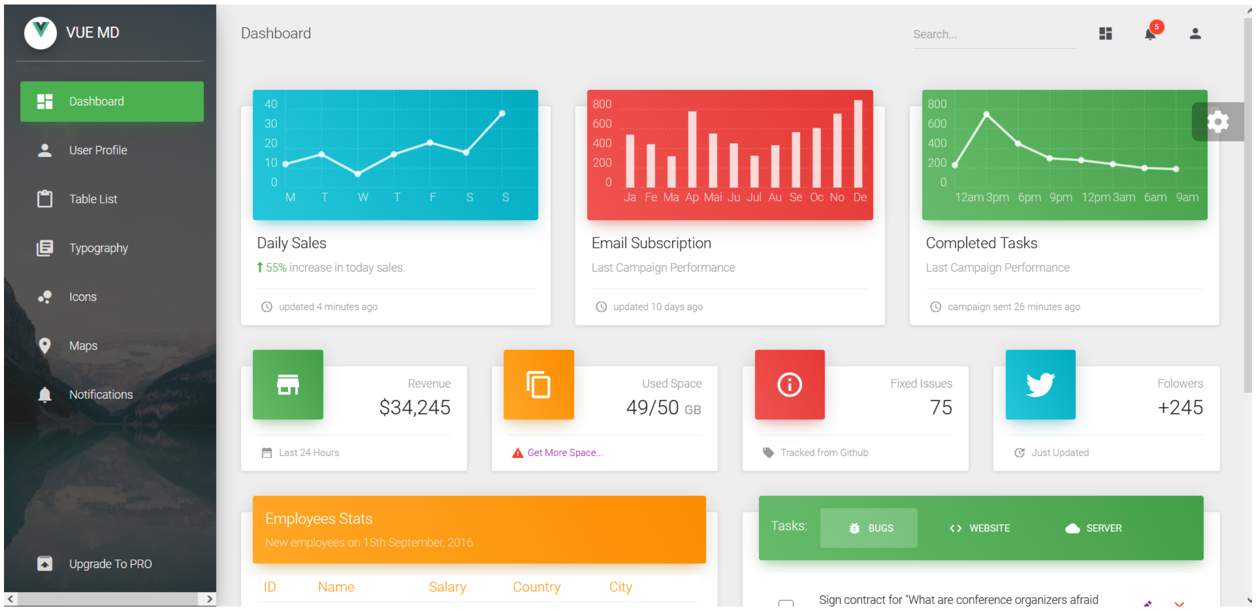
Once the Sensor Node device is configured, it can share the data using MQTT to the server.

The administrator will be able to get an overview of all running nodes in the dashboard hosted on the server.

WebApp

The web app will be made using ReactJS for frontEnd, NodeJS and FastAPI for BackEnd and Mongo DB for the database.

I will use CreativeTim dashboard template to create a nice-looking management portal. A sample preview of the dashboard is shown below.



Please note that the above-shown picture is a template; the dashboard will be designed depending on the project requirements and will not have all the functions as shown in the template.

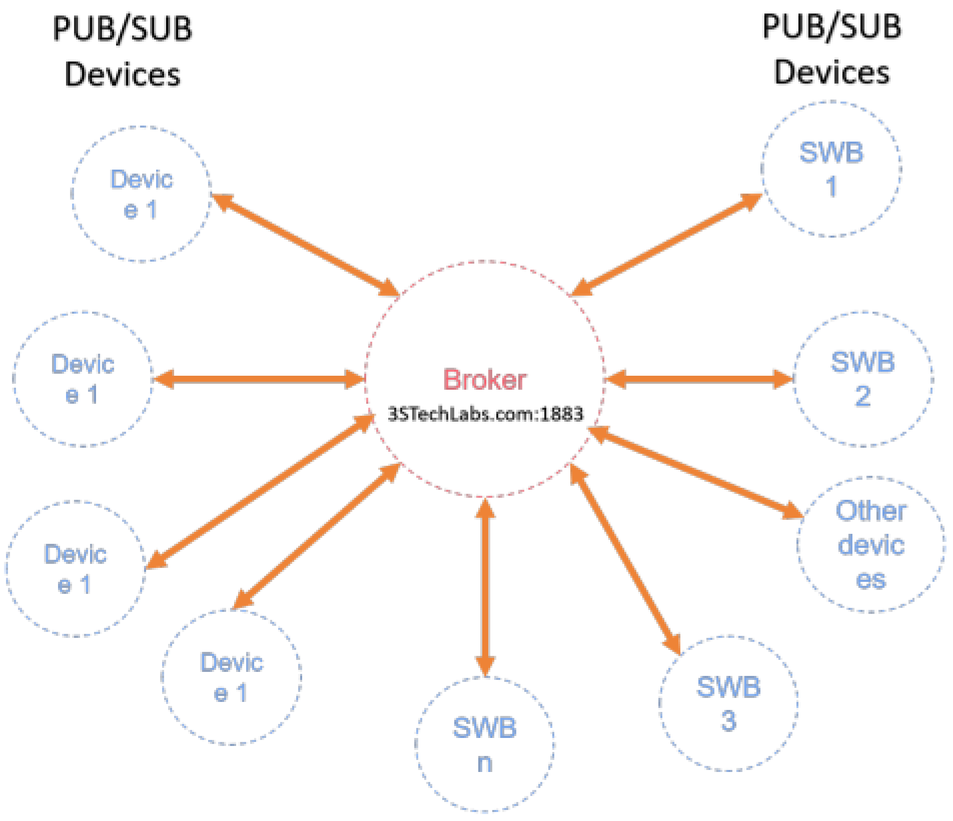
Features of the WebApp

1. Minimalistic design
2. Admin login/signup
3. A global network of devices overview.
4. Adding/controlling and managing the devices.

Smartphone App

The smartphone app will be developed in ReactNative and act as a sensor node. The Smartphone app will be similar to the client WebApp in terms of functionality. Moreover, the smartphone app can provision the device using the CaptivePortal.

What is MQTT?



Take an example of a system in which hundreds of people have smart bands that can display information about a person's surroundings. And then, there are Android, iOS and Windows devices that can be used to monitor smart bands to define a set of parameters for bands.

So in a scenario where there are mixed types of devices including hardware platforms, the best communication protocol is MQTT.

It can handle two-way and parallel communication and the number of devices that can be connected and communicate via MQTT are limitless. The only limit is server resources. MQTT is also known as pub/sub protocol.

Hence the protocol of choice here is MQTT.

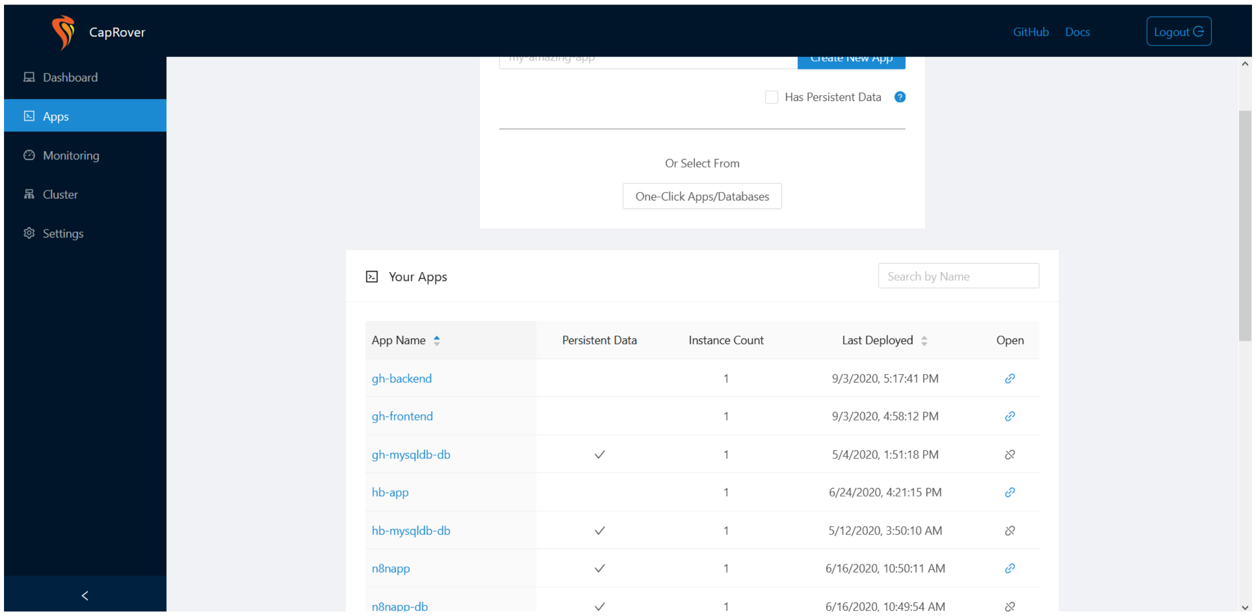
Management and CI/CD Pipeline

We will use Github to deliver the code. The developers from my team will be working on Hardware, Firmware, Backend, Frontend and apps simultaneously and for modularity, we will dockerize different components of the app.

For Continuous Integration and Delivery, we will be using CapRover running on a bare-metal Ubuntu 18.04 instance and each component of the project's GitHub repository will be linked to the respective Containers running in the Caprover. It will allow fully automated delivery.

We will use our own company's production server running on AWS for testing and delivery during the development time. After that we can transfer the files to your own AWS and details of migration are mentioned in the Terms and Condition section of this report.

My automation engineer will work on this.



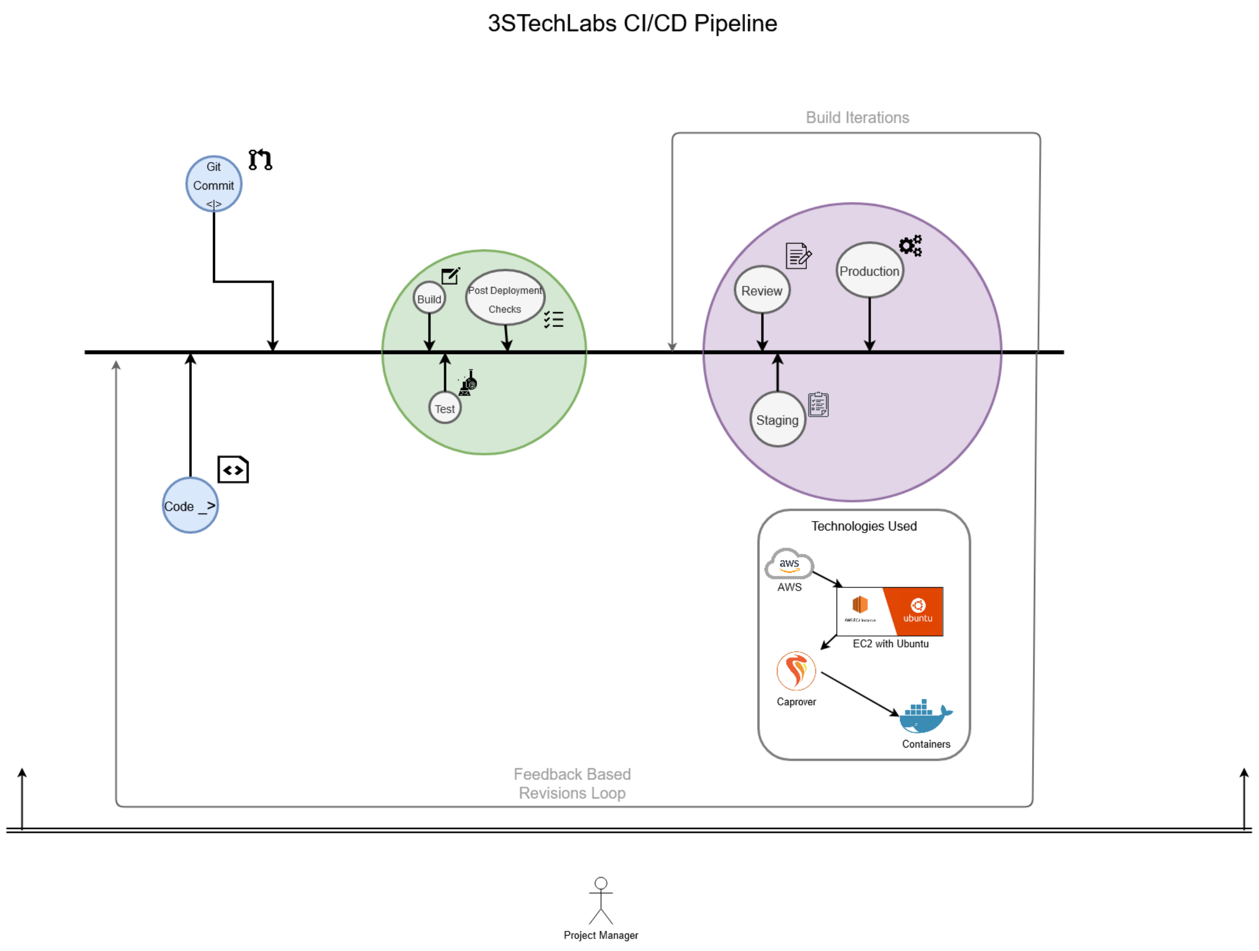
Your Apps

App Name	Persistent Data	Instance Count	Last Deployed	Open
gh-backend		1	9/3/2020, 5:17:41 PM	🔗
gh-frontend		1	9/3/2020, 4:58:12 PM	🔗
gh-mysqldb-db	✓	1	5/4/2020, 1:51:18 PM	🔗
hb-app		1	6/24/2020, 4:21:15 PM	🔗
hb-mysqldb-db	✓	1	5/12/2020, 3:50:10 AM	🔗
n8napp	✓	1	6/16/2020, 10:50:11 AM	🔗
n8napp-db	✓	1	6/16/2020, 10:49:54 AM	🔗

From the picture above, you can see that all of the services like backend, frontend and database etc, are running in separate containers allowing a smooth delivery pipeline.

3STechLabs' CI/CD Standard Pipeline

Below is our well-tested CI/CD Pipeline for project management and delivery. This has been proven to work for our 10+ Full-Stack IoT Projects and Products.



(For a detailed look, please open the image file of above diagram)

Node Casing

Sensor Node in a Case

The casing will be designed in Fusion360 and design will follow your proposed design. It will have ESP32, SSR and Power Module in it.

The picture above is just a sample and we can have a different design as per your requirements.

Questions and Their Answers

1. Could you develop a prototype for us?

- Do you want me to send you the completed prototype to your address? If yes, unfortunately, I can't do this at this moment and neither can I commit anything like that because of the current Covid-19 situation. The flights are delayed, eventually delaying the shipments and, in some cases, even missing the delivery altogether.
- On the other hand, I will provide you
    - Complete Code
    - Circuit diagrams
    - A complete report on how to set up and configure things.
- You can use all the delivered files and documentation to assemble. The prototype yourself.

Milestones Breakdown

1. Sensor Node Design - **Milestone 1** **$300**
    a. (Prototype)Hardware, Schematics, Firmware
    b. Documentation
    c. Prototype using off-the-shelf components(Breadboard-based)
    d. Development Time 15 Days
2. Sensor Node Design - **Milestone 2** **$200**
    a. OTA Functionality
    b. Documentation
    c. Development Time 10 Days
3. **(OPTIONAL)** Smartphone App Design - **Milestone 3** **$500**
    a. ReactNative Based App(for Android)
    b. iOS app source code will be provided as well(no testing on iPhone).
    c. Documentation
    d. Development Time 20 Days
4. IoT Server Design and Configuration - **Milestone 4** **$480**
    a. WebServer backend, database, and other related things + Setup Document
    b. Frameworks: NodeJS, Python
    c. Development Time 15 Days
5. WebApp Dashboard Design **Milestone 5** **$750**
    a. Admin panel and client portal Based on the CreativeTim dashboard.
    b. Frameworks: ReactJS/VueJS
    c. Development Time 20 days
6. Sensor Node PCB Design - **Milestone 6** **$300**
    a. With all STEP, GERBER, BOM and Design Files
    b. Development Time 12 Days
7. **(OPTIONAL)** Sensor Nodes Casing Design - **Milestone 7** **$200**
    a. 3D Printable Casing design using Fusion360(For PCB)
    b. Development Time 7 Days

---

8. Managed Webapp Hosting **$50/month**
    a. On 3STechLabs Production Server
9. After Sales Support/Revisions
    a. Changes in dashboard/backend/database/System Design **$50/hour**
10. **(OPTIONAL)** Webapp Migration **$120**
    a. Migration of the webapp to the customer's server.
    b. The client will provide Ubuntu 20.04 Server, a domain name and a static IP Address.
11. Apple App Store Developer Account **$99+$19.8/year**
    a. 20% Fiverr fee $19.8
12. Android to iPhone app Conversion and app store publishing **$100**
    a. Converting ReactNative android app to iPhone app and publishing it on the iPhone app store.

Terms and Conditions

- We will not ship anything physically unless otherwise decided. Instead, you will be provided with a setup document with the completion of each milestone which will be easy to follow and will contain all of the necessary information.

In this document, the company refers to 3STechLabs.

Please go through the IoT Product Development Contract Devliered separately, sign it and share the signed version with us.

Profile

Name: Nauman Shakir

Company: 3STechLabs

Designation: Founder and Program Manager

Email Address: NaumanShakir3S@gmail.com

Profile: https://NaumanShakir.com

I'm a Full-Stack IoT Developer and have done more than 200 hardware projects and running an IoT and Hardware Design House

Portfolio: https://github.com/Nauman3S

https://3STechLabs.com

https://facebook.com/3STechLabs

https://Linkedin.com/company/3STechLabs

Freelance Profiles

https://www.fiverr.com/naumanshakir

https://www.upwork.com/fl/naumanshakir3s