

EMERGING TECH CONFERENCE – Edge Intelligence

Volume 02, 2023, Page 39 – 42

Proceedings of Emerging Tech Conference:  
Edge Intelligence 2023

S2W-Med Health Status Monitoring

N. Sionas, G. Pomportsis, P. Christodoulou, I. Tzamtzis\*

ELBIS PC, 14th km Thessalonikis – Moudanion ,Thermi – Thessaloniki, 57001, Greece

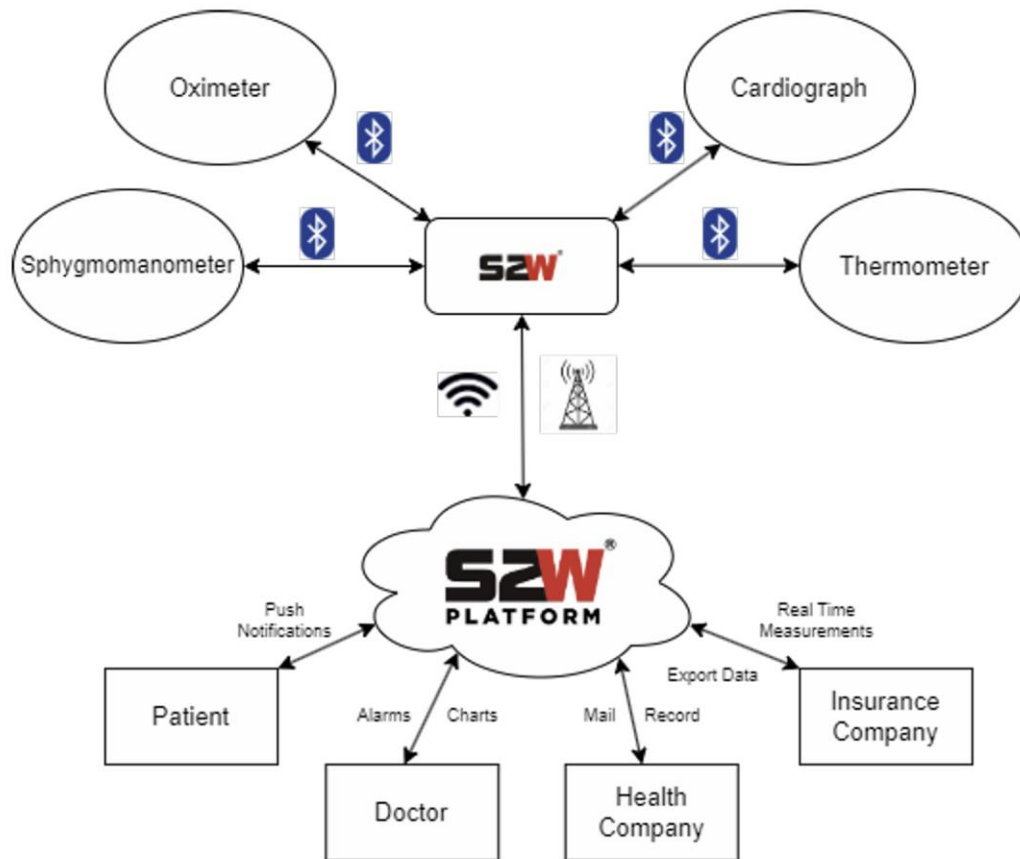
\* Corresponding author. Tel.: +30 216 900 2750, E-mail address: [itzamtzis@elbis.gr](mailto:itzamtzis@elbis.gr)

**Abstract**

It is common for patients who suffer from chronic health problems or are in a period of recovery to need constant monitoring of their body's vital signs by their personal doctor. Such a situation is time-consuming and costly for both the patient and the attending physician. S2W-Med that is presented in this paper is a system for monitoring patient's vital signs in a home environment. Its purpose is to gather the measurements from a group of medical devices so the attending physician has the results available in real time.

**1 Introduction**

The basic system devices are sphygmomanometer, oximeter, cardiograph and thermometer. The devices have the ability to send data via Bluetooth to a central device (S2W). S2W collects the data from all devices and sends them to the cloud via a GSM network. From the API of S2W-Med we monitor the measurements both through the Dashboard S2W application and the mobile application. In essence, Med-S2W upgrades existing tools and gives them IoT capabilities.



Basic Block Diagram

## 2 System description

### 2.1. Cloud infrastructure

The cloud infrastructure implements data management and storage methods and is also responsible for making the data accessible to users through a programming interface (API). The connection of the S2W device to the cloud infrastructure is achieved using a GSM network. The functionality of S2W allows to read values of sensors to which it is connected. To transmit and send data and settings to S2W devices, we use the MQTT protocol. The cloud infrastructure allows us to keep an overview of the device using shadows. Shadows are a representation of the last reported state of the device which includes the last settings we may have sent for the device to receive. Data is sent and received in JSON format. In particular, when sending the data from the device to the cloud, we use the SenML standard, which allows us to limit the size of the data transmitted. The general idea of SenML is that each value included in the JSON object sent is the result of subtracting the value from the first value of the object. The process is followed in reverse when the data reaches the cloud infrastructure.

### 2.2. Backend

We developed the backend in python's Django framework which is responsible for collecting and maintaining sensor measurements.

The backend (API) is a system consisting of smaller subsystems which are: MQTT broker, lambda functions, Time Series Database and HTTPS server. Lambda functions filter and store the data in the Time Series Database, where CRUD (create, read, change, and delete) operations can be performed on it. The HTTPS server application, which is also the system seen from the user application side, can perform operations from the database.

### 2.3. Frontend

It was developed using the ReactJS library. Also, to maintain a global state of the application, the Redux library was used in combination with the Redux-Sagas library. The purpose of the application is to monitor the values of the sensors as well as apply settings to them, this is achieved by calls to API-Django.

### 2.4. Application for mobile devices

A mobile application has been designed for iOS and Android.

Some of its features are:

- Real-time measurements.
- Setting measurement limits for each device
- Indication when exceeding measurement limits on the device
- Device status indication (connectivity, battery capacity, etc)

## 3 Development Tools

The tools used for the project development were:

### 3.1. Software Tools

- ESP-IDF (Programming interface for esp32)
- Django (Backend framework)
- Celery (Task queuing / event transfer from Lambda to Backend)
- Boto 3 (Amazon library for the interconnection of AWS cloud with Django)
- Influx DB (Time series database to store measurements and events )
- Docker (Virtual machine)
- ReactJS (JavaScript framework for frontend app)

### 3.2. Development environments

- Eclipse IDE (Programming environment for writing esp code in C language)
- VS-CODE, (Editor for backend and frontend development)

### 3.3. Infrastructure

- AWS-IOT (Amazon MQTT server for data transfer from device cloud)
- AWS-LAMBDA (Serverless functions for data transfer from AWS-IOT to Backend)
- AWS-RDS (Cloud database to store settings and users)
- AWS-EC2 (Cloud host)

#### 4 Conclusion

The S2W-Med system was created to fill a gap in patient's surveillance.

On one hand, the patient with a small cost feels the security of being constantly monitored by his doctor and even from the comfort of his home, without requiring frequent visits, thus saving time and money.

On the other hand, the doctor has a more complete picture of the patient's condition. Without burdening his time with continuous visits, he receives the indications in his office, automatically creates his patient's record and with notifications and alarms has immediate information about his patient's condition so that he can intervene when necessary.

Also, apart from the four basic devices (sphygmomanometer, thermometer, oximeter, cardiograph), new medical devices are being tested and integrated in the S2W-Med system in order to make the surveillance of a patient's condition more complete and also become a monitoring tool for more medical specialties.

#### 5 Conclusions

With this work we presented a rapid prototyping approach. We use MATLAB-Simscape to insert the physical and electrical model of the cells in a specific configuration to form the battery module. We model the rest of the battery configuration by modeling the BMS functions as well as the battery charger behavior. By simulating the whole system, we can extract its behavior and use it to carefully size the battery components. We present the final step of this process by designing the key aspects of the battery with CAD software.