

Feature Description Document (FDD)

Version 1.0

**Table of Content**

[**1.0 Feature Description**](#_34q3j3duernk) **3**

[1.1 Assumptions](#_bzcze5tfj7cr) 3

[1.2 In-Scope](#_kxaodviixocn) 3

[1.3 Out-of-Scope](#_o2t1jxq2lkoe) 3

[1.4 Future Sope](#_8z684xbv6nie) 3

[**2.0 Database Schema**](#_6w6kxbbmaxnn) **3**

[2.1 End-to-end Flow](#_ubxo3s9rzsg4) 4

[**3.0 Solution Architecture**](#_7sxr3hqmod09) **5**

[**4.0 Deployment Strategy**](#_atwvamtx0gpf) **6**

[4.1 Benefits](#_ah29o0x1w1az) 6

[**4.0 Technology Stack**](#_5oqn9k1xrpl5) **6**

# **1.0 Feature Description**

Develop a web service, using resources on AWS, that accesses the database in real-time and provides excellent visualization of the time series analysis of Voltage, Current, and Power (Voltage \* Current).

## 1.1 Assumptions

* AWS account shall be used by the developer.
* All the requirement is shared through the mail.

## 1.2 In-Scope

* Create voltage and current curve of the selected sensor ID.
* Create a power curve of the selected sensor ID.

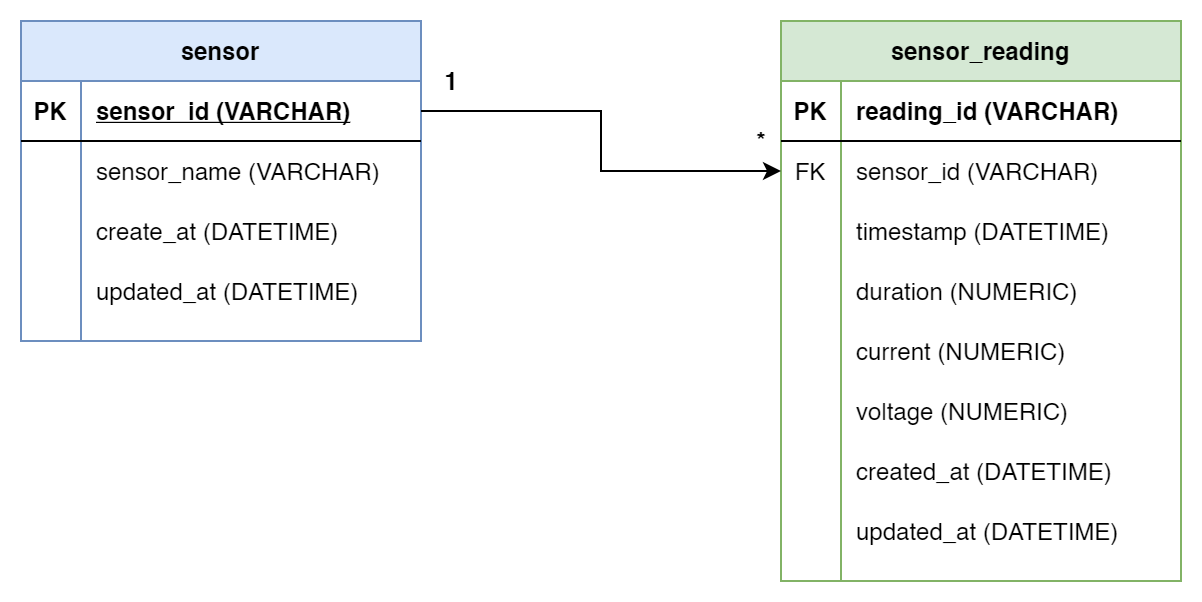
## 1.3 Out-of-Scope

* Adding a new record to sensor ID.

## 1.4 Future Sope

* User Authentication.
* Role Base Access Control of the application.
* Near to real-time notification service.

# **2.0 Database Schema**



## 2.1 End-to-end Flow

The end to end flow will be following for the feature -

* We will populate the data from provided datasheet into the database using the automation scripts.
* For populating the data to the frontend, the user will first select the sensor for which the data needs to be populated.
* Backend will populate the sensor\_reading data to the end-user based on the Sensor selected from UI.
* The backend server will return the sensor readings back to the frontend.

# **3.0 Solution Architecture**

# 

# **4.0 Deployment Strategy**

## 4.1 Benefits

* Scaleable
* Easy Deployment
* Cost-Efficient

# **4.0 Technology Stack**

* AWS
  + AWS Lambda
  + AWS RDS
  + AWS API gateway
  + AWS S3
* Python 3.x
* Serverless Framework
* UI
  + HTML, CSS and Javascript