Faculty of Engineering - Helwan University

Computer and Systems Engineering Department

Electronics and Communications Engineering Department

NEXT-GENERATION SMART GRID METER (NGSM)

Prepared By:

Abdelrahman Adel Ibrahim

Abdelrahman Osam Khaled

Abdullah Osama Mohamed

Belal Abdelaziz Abdullah

Noor El-Deen Magdy Mohamed

George Attia Nafady

Mai Hesham Ahmed

Hossam Mamdouh Hossam

Rania Mostafa Mohamed

Supervised By:

Assoc. Prof. Amr Elsayed

Assist. Prof. Mohamed El-Dakroury

Graduation Project Thesis for Completing Bachelor's Degree in Computer and Systems and Electronics and Communications at Helwan University, Cairo, Egypt

2021/2022

Acknowledgment

First and foremost, we are grateful to our great creator Almighty Allah for the good health and wellbeing that were necessary to complete this project.

It's our proud privilege to release the feelings of our gratitude for the immense contribution of these people who directly or indirectly made this project a success.

We would like to express our sincere gratitude for our professors who have shown righteous disposition, brilliance and great intellect while valiantly supporting us with their time, knowledge and experience for us to successfully commence, manage and conclude our project in a noticeably enlivened state.

We would like to express our hearts full indebtedness and owe a deep sense of gratitude to our graduation project supervisors **Assoc. Prof. Amr Elsayed and Assist. Prof. Mohamed El-Dakroury** for their advice and knowledge and the insightful discussions and suggestions throughout the whole venture, and for patiently guiding us through its processes.

We would like to thank **Eng. Abdalla Ali, Eng. Omar Mekkawy and Eng. Ibrahim Hamoda** and all of the **Iskraemeco R&D team** for their dedication, their efforts and their great support to our team.

You made us believe that we have what it takes and we have the potential to reach greater heights in our graduation project.

Finally, we would like to thank **Iskraemeco** for all the efforts and help to ensure that our graduation project meets how they work and meets the qualifications of modern technologies.

We end this where it all started with loving and supporting families, where our strength, courage and confidence originate. We extend our gratitude and appreciation to our parents for their affection and sacrifices through our life.

We are indebted to your everlasting support.

Abstract

Next-Generation Smart Grid Meter (NGSM) is a smart meter device for monitoring electricity consumption. It measures the energy consumption of a household and also acts as a smart node in the electricity grid to apply the next generation of Advanced Metering Infrastructure (AMI) systems connected via a smart grid for cluster communication. Various features would be included such as measuring the power, energy...etc., calculating the costs with a developed billing system connected to a web-based system for the overall monitoring across the supported region, and displaying the readings on a TFT screen for the user, and tamper detection mechanisms. The smart grid connection would be via the internet which transmits the data over the grid, moreover it applies a Machine learning model for using the data collected in its data storage to make classification and prediction of the working household appliances at different times. The concluded NGSM system is implementing a complete advanced metering infrastructure (AMI) that includes an Embedded Linux based meter for on-edge processing with a complete Head-End System and Mobile application . Appliance detection using the aggregated load achieved 99% accuracy with 3 devices operating on the same time. Energy demand forecasting of a collected smart meter data to predict the next 6 months from 3 years of data.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Definitions

AC - Alternating Current
ADC - Analog-To-Digital Converter
AAL - Ambient Assisted Living
ANN - Artificial Neural Network
AMI - Advanced Metering Infrastructure
APN - Access Point Name

BLUED - Building-Level fUlly labeled Electricity Disaggregation dataset
BER Bit Error Rate

CDM - Committee Decision Mechanism
DC - Direct Current
COSEM - Companion Specification for Energy Metering
CSS - Cascading Style Sheets
CPU - Central Processing Unit

DFT - Discrete Fourier Transform
DUT - Device Under Test
DHCP - Dynamic Host Configuration Protocol
DIN - Deutsches Institut für Normung
DLMS - Device Language Message Specification
DNS - Domain Name System/Server/Service

EMC - Electromagnetic Compatibility
EMI - Electromagnetic Interference
EMIL - ElectroMagnetic Interference for Load disaggregation dataset
EAL - Evaluation Assurance Level
EAP - Extensible Authentication Protocol
EN - European Norm
ESD - ElectroStatic Discharge

FFT - Fast Fourier Transform
FHMM - Factorial Hidden Markov Model
FSM - Finite State Machine
FCC - Federal Communications Commission
FEM - Field Exchangeable Module

FIFO - First In, First Out
FLASH - Electronic (solid-state) non-volatile computer storage medium that can be electrically erased and reprogrammed.
FQDN - Fully Qualified Domain Name
FW - FirmWare

GAAC - Group-Average Agglomerative Clustering
GLR - Generalized Likelihood Ratio
G3-PLC Communication technology with low layer protocol for enabling large-scale infrastructure on the
electrical grid
GND - Ground
GMK - Grand Master Key
GPRS - General Packet Radio Service
GRE - Generic Routing Encapsulation
GSM - Global System for Mobile
GW - Gateway (AC550)
GUI - Graphical User Interface

HAC - Hierarchical Agglomerative Clustering
HMM - Hidden Markov Model
HVAC - Heating, Ventilation and Air Conditioning
HES - Head-End System
HTML - HyperText Markup Language
HTTP - Hypertext Transfer Protocol
HTTPS - HyperText Transfer Protocol Secure
HW - HardWare

I2C - Inter-Integrated Circuit
ICCID - Integrated Circuit Card IDentifier
ID - IDentification
IEC - International Electrotechnical Commission
IMEI - International Mobile Equipment Identity
IP - Internet Protocol
IPv4 - Internet Protocol version 4

LAN - Local Area Network
LED - Light Emitting Diode

Linux - free and open-source software operating system
LTE - Long-Term Evolution

kNN - k-Nearest Neighbor

MAD - Mean Absolute Deviation
MAC address Media Access Control address

NGSM - Next-Generation Smart Grid Meter
NILM - Non-Intrusive Load Monitoring
NC - Not Connected
Node - In Web user interface means attached device
NTP - Network Time Protocol

OS - Operating System
OVC - OverVoltage Category

PLC - Power Line Communication
PSK - Pre-Shared Key
PFC - Power Factor Correction
PFM - Pulse-Frequency Modulation
PWM - Pulse-Width Modulation
P2P - Peer-to-peer
PAN - Personal Area Network
PIN - Personal Identification Number
PPE - Personal Protective Equipment
PUK - PIN Unlocking Key
PW - password

RAM - Random-Access Memory
RSSI - Received Signal Strength Indicator
RTC - Real Time Clock
RX - Receive
REDD - Reference Energy Disaggregation Dataset
RMS - Root Mean Square
SoC - System on Chip
SPI - Serial Peripheral Interface
SIM - Subscriber Identity Module

SW - SoftWare
STFT - Short-Time Fourier transform
SMPS - Switched-Mode Power Supply
SOM - Self-Organizing Maps
SVM - Support Vector Machine
TCP - Transmission Control Protocol
TLS - Transport Layer Security
TPM - Trust Platform Module
TX - Transmit

UART - Universal Asynchronous Receiver-Transmitter
UK-DALE - UK Domestic Appliance-Level Electricity dataset

VSD - Variable Speed Drive

WAN - Wide Area Network
Web service - A method of communication between two electronic devices over the World Wide Web.
Web-UI - Web User Interface

Team Contacts

SN	Name	Role	Email/Number
1	Noor El-Deen Magdy Mohamed	Artificial Intelligence and Embedded Linux	nooreldeenmagdy@gmail.com / +201150462982
2	Abdalla Osama Mohamed	Back-End Developer	abdullahelagami55@gmail.com / +201092272796
3	Abdelrahman Adel Ibrahim	Artificial Intelligence and Embedded Linux	abdulrahman.adel098@gmail.com / +201146631026
4	Mai Hesham Ahmed	Front-End Developer	maihesham789@gmail.com / +201028520655
5	Rania Mostafa Mohamed	Mobile Development	raniamostafah55@gmail.com / +201117199897
6	George Attia Nafady	Embedded Linux and Hardware	johnatia97@gmail.com / +201007629932
7	Abdelrahman Osam	Embedded Linux and Hardware	abdelrahmanosam7@gmail.com / +20 101 785 3167
8	Belal Abdelaziz	Embedded Linux and Hardware	belalabdalaziz@gmail.com / +201122548571
9	Hossam Mamdoh	Embedded Linux and Hardware	doo7a7420@gmail.com / +201111520654

1. Introduction

1.1 Problem Statement

Electricity is not a free resource and it is an important national treasure. Electricity generation and its distribution are a major concern and need to be properly handled by governments. Outdated infrastructures, insufficient resources, a lack of integrated communication platforms, cost of deployment, the transition from legacy systems, management of vast amounts of data, compatibility of older equipment, a lack of standards, and changes in regulatory protocols and policies are all reasons why smart electricity metering solutions are not widely adopted.
Electricity meter technology advancements in recent years were essential for monitoring and managing electricity consumption as the electricity generation sector in Egypt is responsible for approximately 40% of the country's total CO2 emissions. So there was a huge need for replacing the traditional electricity meters with smart electricity meters for more accuracy, efficiency, features, privacy, security, and visibility in electricity measurements. Thus, enabling a better understanding of energy usage patterns and the ability to improve customer experience, consumption monitoring through web and mobile management systems, grid reliability, outage management, and operational efficiency with the ML bills and household appliances prediction for usage recommendations. **[1]**

1.2 Problem Importance

- Improve energy habits.
- Avoiding the capital expense of building new power plants.
- Reduce carbon emissions and the load on the electricity power plants.
- Track and compare the energy usage.
- No need to manually submit electricity readings.
- Automatic and accurate meter readings with electricity usage recommendations. **[2]**

1.3 Current Electricity Meters Solutions

In recent years, domestic and industrial users have shifted from traditional meters to smart meters. Electromechanical meters were a dominant part of electricity measurement before 1970. They could only measure the electrical energy. However it had been identified that the requirement of a meter which could communicate and measure the electrical energy along with other electrical parameters. Therefore solid state electronic meters were introduced to measure the overall electrical parameters.
Between 1970 and 2000, automatic meter reading was added to electronic meters and it was a great achievement since it could send the data in near time. However it could only provide one-way communication. This limitation was overcome by the introduction of smart meters which can provide two-way communication. Smart meters can measure all the electrical parameters like electronic meters and communicate data in a meaningful way. The consumer is updated with electricity usage, cost, tariffs and other notifications sent by the utility. Smart meters have different functionality to manage the end user loads and run them in an optimal way to reduce the electricity bill as well as to conserve energy.**[3]**

Now, Let’s show some steps that the meter had taken to reach the smart case:

1.3.1 Electromechanical Energy Meter

Electromechanical energy meter is the most traditional and widely used energy meter over a century. It is capable of measuring only the active energy which is typically displayed on a mechanical counter in kWh.

Figure 1.1: Electromechanical meter

How does it work?

It is basically designed with four major systems which are the driving system, moving system, breaking system, and registering system. The driving system consists of two electromagnets while the moving system consists of an aluminum disc. The permanent magnet acts as the braking system while the gear train and counter act as the registering system. The electromagnetic force is produced by the arrangement of voltage and current coils. The voltage coil is connected across the supply while the current coil is connected in series with the load . The voltage coil produces a magnetic flux in proportion to the voltage and the current coil produces a magnetic flux proportional to the current. The aluminum disc is mounted on a rigid axis. A mechanical force is exerted on the disc by the Eddy currents produced. The register mechanism integrates the speed of the disk over the time by counting the number of revolutions.
Current coil or the series coil produces alternating flux which is proportional and in phase with the load current. Voltage coil or the shunt coil carries a current proportional to the supply voltage. The flux produced by the voltage coil is not in phase with the supply voltage. This flux is 90 degrees lagging with the supply voltage. This is done by having properly adjusted copper rings in the flux path However some electromechanical meters use winding with the series connected lag adjusting resistor to perform this task.

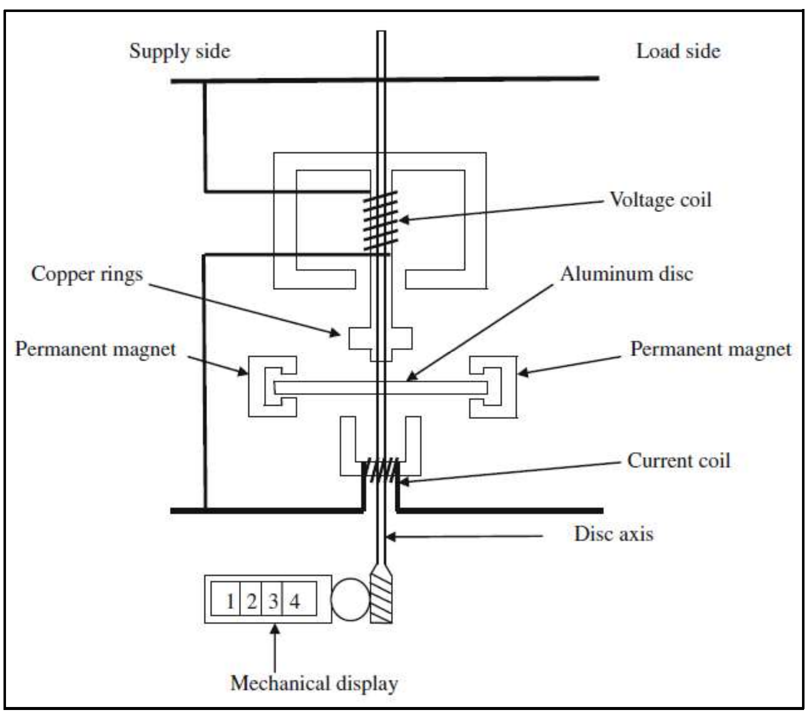


Figure 1.2: Components of a single phase electromechanical energy meter

Drawbacks:

- Electromechanical meters react to the changes more slowly than digital meters. They have many susceptible errors due to environmental variations and regular operations.
- The moving parts inside these meters are prone to wear over time, varying temperature, and conditions.
- On the other hand mechanical gears wear due to effects of dirt, dust and humidity. The gear ratios also change over time due to lack of lubricants.
- Vibration and shock affect the accuracy of the meter in the long run.
- Due to the lack of linearity of iron core and the inertia of the spinning disk, errors can be caused at low and high loading
- Electromechanical meters require manual readings. In other words meter readers have to go and take the reading manually to issue the bill.
- Because of the man power requirement, there is always an additional cost to the bill apart from the energy consumed.
- The tampering of meter readings, human errors in readings, irregularities in billing time, and controversial billing are also possible with manual readings.
- Power theft is a major problem caused with electromechanical meters. Illegal reconnection of power lines, bypassing of the energy meter, and very weak conditional access enforcement cannot be detected directly with these meters.

1.3.2 Electronic Meter

Figure 1.3: Electronic meter

How does it work?

An electronic meter meets the basic requirement of measuring the amount of electricity being consumed by an account. A representative from the utility provider must physically examine the meter and obtain a reading at the time of bill generation. If a visit is not possible, the utility company will have to issue an estimate-based charge. Both the consumer and the corporation may suffer as a result of this. Furthermore, typical traditional meters do not record any information. Traditional electricity meters only show the quantity of electricity consumed since someone (meter reader) last read the meter.

Drawbacks:

About the traditional meters:

- No data storage
- A meter reader physically comes to the customer's home or business to record the information and send it to the metering company
- If access cannot be gained to the meter, this may result in estimated bills
- Electricity use is tracked by either waiting for a customer's monthly or quarterly bill or manually reading customer's household meter by oneself
- No outage detection, as distribution companies cannot react quickly to interruptions in the supply
- Connections and disconnections must be done manually.

1.3.3 Smart Meters

A smart meter measures and records energy use data in the same way that a standard meter does in your home. The smart meter, on the other hand, is a digital gadget that can communicate with your utility remotely. It eliminates the need for a meter reader by sending your consumption data to your utility every 15 minutes to an hour.

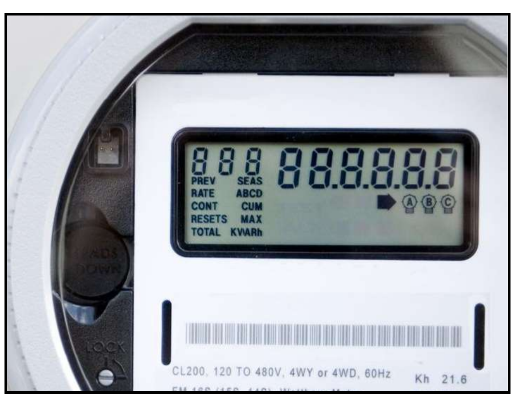


Figure 1.4: Smart meter

Smart metering helps:
- Electric utilities eliminate manual monthly meter readings.
- Avoid capital expenses of building new power plants.
- Optimize income with existing resources.
- Dynamic pricing, which raises or lowers the cost of electricity based on demand.
- Power resources are used more efficiently.

Smart metering helps electricity consumers:
- Adjust their habits and lower electricity bills.
- Reduce the number of blackouts and system-wide electricity failures.
- Obtain greater and more detailed feedback on electricity usage.

Smart metering helps the environment:
- Reduce the need for new power plants, which produce greenhouse gasses. (GHG) that substantially creates pollution resulting in health risks.
- Curb existing GHG emissions from existing power plants.

About smart meters:
- Data storage stores electricity consumed.
- Data is automatically transmitted to the metering company.
- Digital data of energy consumption and you are provided in near real-time.
- Automated outage detection enable distribution companies to restore power quicker than traditional electricity metering.
- Connections and disconnections are faster because they are managed remotely.

The advantage of applying smart meters:

The smart meter can inform the utility immediately if there’s a power outage in your area. smart meters can quickly dispatch crews to resolve the situation and get your power back on as soon as possible. Once everything is back to normal, the smart meter will notify your utility of the resolution.

The ability to track energy usage is one of the most significant advantages of a smart meter for consumers. Most smart meters include a digital face that displays real-time data on how much energy you've consumed. Though it won't tell you what in your home consumes the most electricity, knowing how much energy you use can help you make adjustments.

Another advantage that smart meters have over traditional meters is with respect to energy efficiency. These meters store data and make it accessible to the company as well as customers. By accessing this data, a clearer understanding of consumption patterns becomes visible. Based on this data, the company can make decisions for better utilization of their resources and customers can get detailed feedback on their electricity usage.

Some state governments or utilities even offer online programs that allow you to track your energy consumption data. The Texas government, for example, offers a program called Smart Meter Texas which allows residential and business customers the ability to track energy consumption data in 15-minute intervals, as well as monthly and daily usage information.

1.4 Motivation

Our main motivation is to help as many as possible persons in their lives, and we believe that if You have a chance to save lives If you don’t take it, you may regret it. We should never forget that God granted us the power to reason so that we would do work here on Earth - so that we would use science to save lives, and give hope.

The project motivation points can summarized as follows:

- Smart meters are an opportunity to motivate households for energy savings.
- Providing customers with greater control over their electricity use when coupled with time-based rates.
- Allowing customers to make informed decisions by providing highly detailed information.
- Helping the environment by reducing the need to build more power plants, or avoiding the use of older, less efficient power plants as customers would lower their electric demand.
- The need for energy savings by Improving cash flow and customer service with more timely billing.
- Obtain better information for asset optimization and management:

- Highlights faulty appliances.
- Shows which appliances consume more power.
- Gives the user the recommendations for the appliance's operation.

- Increasing privacy because electricity usage information can be relayed automatically to the utility for billing purposes without on-site visits by the utility collector. **[4]**

1.5 Proposed Project Idea

NGSM is a smart meter device for monitoring electricity consumption on edge. It works as a meter to measure the energy consumption of a household and also acts as a smart node in the electricity grid to apply the next generation of AMI (Advanced Metering Infrastructure) systems connected via a smart grid for cluster communication. Various features are included such as measuring the power, energy...etc., calculating the costs with a developed billing system connected to a web-based system for the overall monitoring across the supported region, and displaying the readings on an LCD screen for the user with tamper detection mechanisms. The smart grid connection would be via the internet which transmits the data over the grid, moreover it applies a Machine learning model for using the data collected in its data storage to classify and predict the working household appliances at different times. **[5]**

Figure 1.5: /root Logo

1.6 Objectives

This point will discuss the basic objectives of our smart meter which are:

1- A smart meter that implements the basic functionalities of an AMI (Advanced Metering Infrastructure), which are:

- Current and voltage measurements.
- Cost, bill calculation, and online payment.
- Displaying electricity data on an LCD screen for the user.

Figure 1.6: AMI

2- Implementing a Linux-based embedded system with a powerful-enough system on chip (SoC).

3- Designing a system with two-way communication between the end-user and the head-end system, which is achieved using:

- NGSM Head-End System.
- NGSM Smartphone Mobile application.
- NGSM Web-App.

4- Making use of smart meter data by applying data analytics techniques, visualization, and machine learning to implement features such as:

- Household Appliances Detection.
- Tampering Detection.
- Energy Demand Forecasting.

- Harmonics Detection.

5- A stand-alone system that is capable of directly communicating with the head-end system without the need for a Data Concentration Unit (DCU).

6- Making enough versions of the said system with which we can create a grid network-based infrastructure to apply the next-generation of AMI (Advanced Metering Infrastructure) systems connected via a smart grid for cluster communication.

7- As a result of the classification, HES will be able to detect tampering, as based on the collected and classified data, HES will know if the consumption has been higher than the usual, by then, it will detect that someone else is tampering and robbing the real owner of the meter.

8- In addition, Harmonics detection is achieved too, as it is supposed that the relation between current & voltage of each appliance be in a form of sine wave, when waveforms deviate from a sine wave shape as shown in **Fig 1.7**, you should know that it includes harmonics, these harmonics affect the quality of the power system, increase heat in conductors, require more power and more cost.

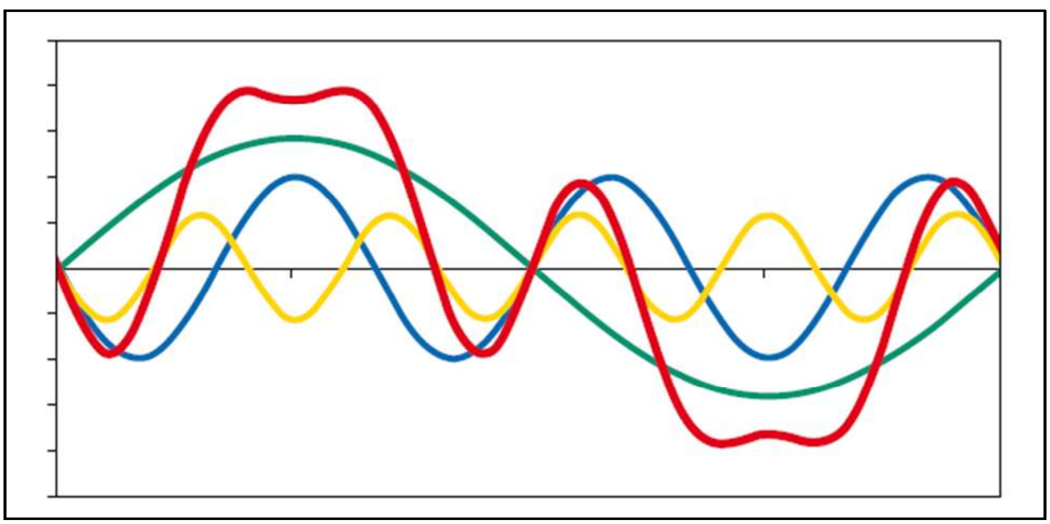


Figure 1.7: Harmonics Detection

9- Based on the classification and analyzing the stored data from the previous year, HES will have the ability to predict the high demand for electricity during a certain time of the day which is called the “rush hour”.

1.7 Educational Value

Working on this project has added new values not only to our technical overview, but also to our non-technical overviews that has helped us as a team and as individuals.

1.8 Technical Value

Embedded Linux Systems
PCB Design
System Architecture and Design
Electricity Metering Technologies
Cloud Computing
Database
UI/UX Design
Web Applications
Mobile Applications
On Edge Machine Learning Deployment

1.9 Non-Technical Value

Idea generation techniques:

1. Writing a business plan.
2. Work division and integration.
3. Time management.
4. Documenting and presenting a large-scale project.

1.10 Domain of the Application

Our system needs cooperation between our NGSM and local authorities for each country to have a standard global system, and with all collected information, system improvement and algorithms enhancement will be easier and faster.

1.11 Related Work

- **Implementations Classification by real commercial examples:**

A. Gomelong

Gomelong is a Chinese Trading Company, Gomelong was established in 2010. It works in the manufacture of energy meters and digital meters . it employs 51 to 100 people
Multi-tariff STS Single Phase Prepaid Electric Meter 20 Digits Token Energy Meter made by Gomelong .

Figure 1.8: Gomelong

Quick Details:
Place of Origin: Zhejiang, China
Phase: Single Phase
Output Voltage: 110V,220V,230V,240V
Operating Temperature: -20-50℃
Dimensions: 169mm*110mm*62.5mm
Type: STS Prepayment Meter
Display: LCD
Frequency: 50HZ/60HZ
meter
Rated current: 5(60)A
Price :16$

Feature:

- Payment by 20 digits token
- Alarm when low energy/overload/fraud.
- With optical communication functions for reading meters and setting parameters.
- Open terminal cover detection alarm.

B. Meatrol

ME437 multi-function WIFI meter single phase smart electric meter/remote stop power meter made by Meatrol

Quick Details:
Place of Origin: Shanghai
Phase: three phase
Support Extra sensor: 333mV CT Rogowski coil
Programmable digital output: Relay
I/O function: 1*digital output
Power: 85~265V AC/DC
Storage: 1GB SD card(Max 4GB)
Display: 35 inch TFT screen display
Sampling rate: 8k samples per second

Figure 1.9: Meatrol meter

Weight: L*W*D:96*96*99mm
Price: **$85.00**

C. Toky

DR9 Series Smart Meter Rs485 Electric 2 Alarms Multi Function Meter Power Meter made by Toky

Quick Details:
Place of Origin: Guangdong, China
Phase: Three Phase
Measuring Energy Range: 6A *480V direct input
Output Voltage: 100V or 450V or others
Operating Temperature: -10--50 Degree
Dimensions: 96mm * 96mmr
Display: Digital LCD Display
Rated current: 5 A ac/1 A ac
Measuring frequency range: 45Hz~65Hz
Power supply: 100~240Vac,50/60Hz
Analog output: 4~20mA
Features: High Reliability
Weight: 1kg
Price: **$275.00**

Figure 1.10: Toky meter

Feature:

- Strong anti-interference ability : ensure the long term stable measurement and safety and reliability in various electric fields
- 28 parameters measurement : single phase and three phase parameters display and record
- Rich screen display : data total display screen, voltage, current, frequency, power active and reactive power.

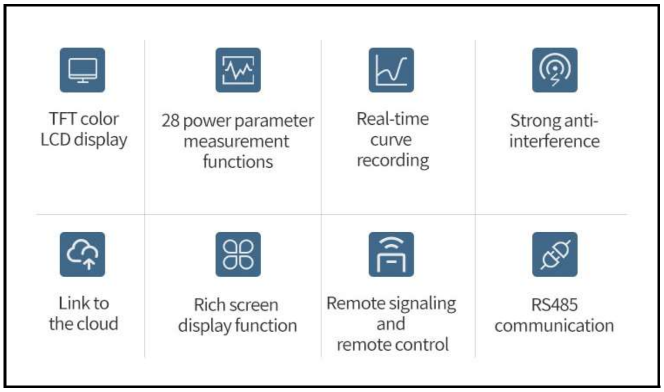


Figure 1.11: Toky meter feature

Implementations Classification by Research Projects:

A. Research project in Qatar university

This research project presents the design of a modular and open-source smart meter for educational and research purposes. The smart meter monitors the energy consumption, uploads the data to a cloud server, and can be used to control appliances remotely through a control relay. Using the Internet of Things (IoT) technology.

They use:

- Arduino uno board
- voltage and current transducers
- Solid State Relay module
- LED Screen
- ESP8266 Wi-Fi Module
- Linear Voltage Regulator

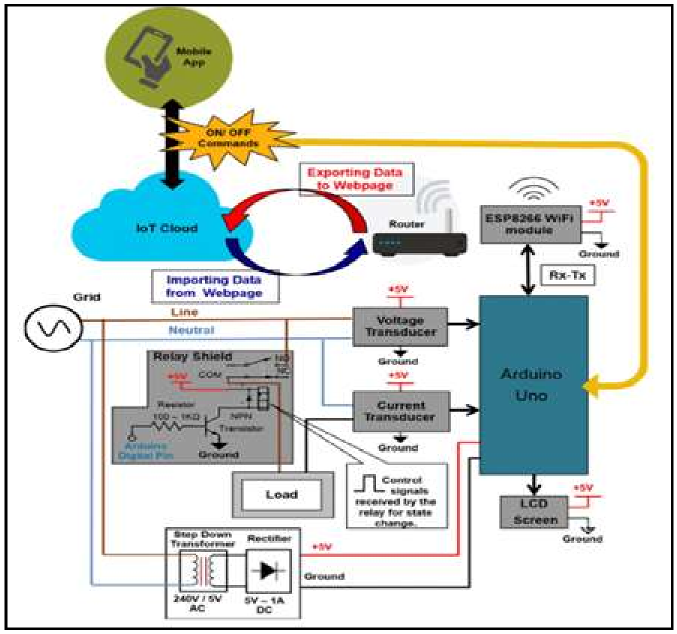


Figure 1.12: Overall smart meter model

B. Research project in ECE, K L University in India

This research project presents the Smart Electric Meter using LoRa works with the wireless data protocols over a long range so, there will be updated readings, no need for the electricity department man to come and check the meter readings in the premises of the user. although there are any delays in the payment, the system will automatically disconnect from the service of the consumer and after the payment the service will send to the consumer, so there is an advantage of not getting counted the cut-off energy and after the connection is established it sends the on power utilized energy. **[6]**

They use:

- NodeMCU (ESP8266) as a Microcontroller
- Lora Module
- voltage and current transducers
- DHT11 Sensor LED Screen
- Buzzer

- Overall smart meter model

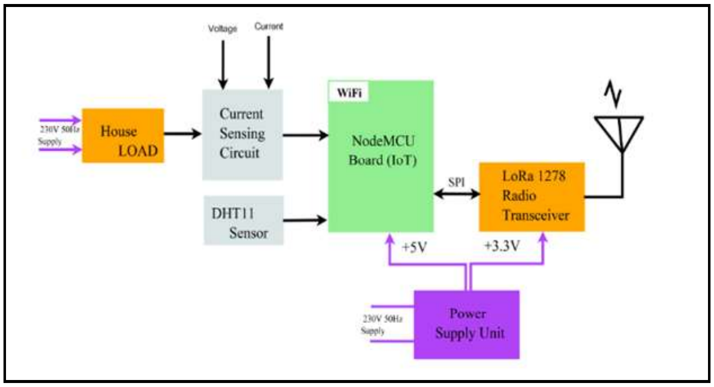


Figure 1.13: Block diagram of transmitter circuit

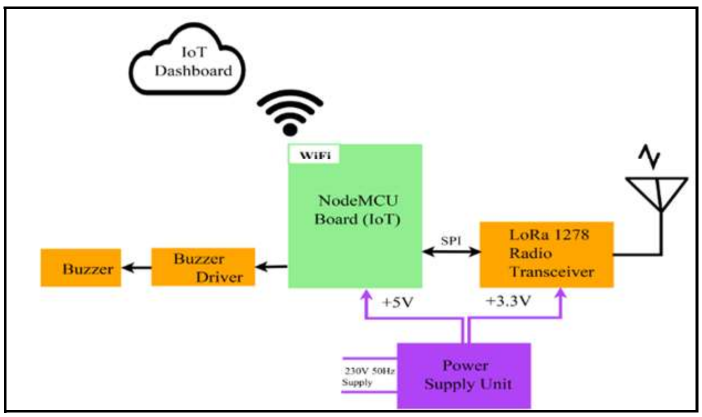


Figure 1.14: Block diagram of receiver circuit

1.12 Thesis Organization

The rest of the thesis is organized as follows. Section 2 introduces the required Technical Background. Section 3 illustrates the NGSM System Architecture. Section 4 talks about the 1st generation of our NGSM meter. Followed by Section 5,6 for the next two generations. Section 7 introduces the NGSM backend system design. Section 8,9 talks about the NGSM Web-App and NGSM Mobile Application. Section 10 shows the ML techniques as Appliance Detection and Energy Demand Forecasting. Section 11 covers the System Modeling part. Section 12 Concludes the thesis and talks about the future work. Section 13 finally shows our References.

2. Technical Background

2.1 Smart Grid

- Overview

A smart grid is an electricity network based on digital technology that is used to supply electricity to consumers via two-way digital communication. This system allows monitoring, analysis, control and communication within the supply chain to help improve efficiency, reduce energy consumption and cost, and maximize the transparency and reliability of the energy supply chain. The smart grid was introduced with the aim of overcoming the weaknesses of conventional electrical grids by using smart net meters.**[7]**

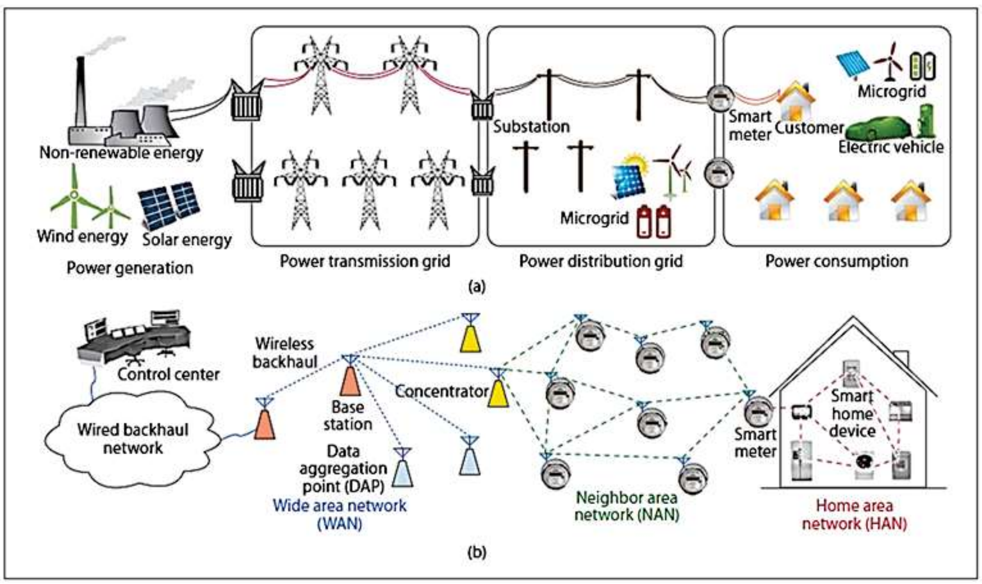


Figure 2.1: Complete layered architecture of smart grid

- What makes "Grid" smart?

In short, the digital technology that allows for two-way communication between the utility and its customers, and the sensing along the transmission lines makes the grid smart. Like the Internet, the Smart Grid will consist of controls, computers, automation, and new technologies and equipment are working together, but in this case, these technologies will work with the electrical grid to respond digitally to our quickly changing electric demand. **[8]**

- Benefits

A smart grid serves several purposes and the movement from traditional electric grids to smart grids is driven by multiple factors. These include the deregulation of the energy market, evolutions in metering, changes in the level of electricity production, decentralization (distributed energy), the advent of the involved ‘prosumer’, changing regulations, the rise of microgeneration and (isolated) microgrids, renewable energy mandates with more energy sources and new points where and purposes for which electricity is needed (e.g. electrical vehicle charging points).

The benefits associated with the Smart Grid include:

- More efficient transmission of electricity.
- Quicker restoration of electricity after power disturbances.
- Reduced operations and management costs for utilities, and ultimately lower power costs for consumers.
- Reduced peak demand, which will also help lower electricity rates.
- Increased integration of large-scale renewable energy systems.
- Better integration of customer-owner power generation systems, including renewable energy systems.
- Improved security. **[9]**

- Smart Grid vs. Traditional Electricity Grid

Traditional electricity grids have almost no storage capabilities, they are demand-driven and have a hierarchical structure. In an electricity network voltage level is gradually changed (up and down) so the electricity can be used by these different consumers: from transmission voltage levels to distribution voltage levels to service voltage levels.
An electrical grid's purpose is to ensure that electricity is always provided when and where needed, without interruption – and herein lie many challenges where a smart grid can already offer solutions/answers.
Given the complexity and the multiple challenges that can arise such as the consequences of severe weather conditions, damage by wildlife, human sabotage and other external factors and internal factors (issues with equipment failure and crucial assets) managing a grid is very complex and a

dedicated field for experts who also need to consider the choices regarding energy regulations and sustainability initiatives by governments.
The two-way flow of electricity and data that is the essential characteristic of a smart grid enables to feed information and data to the various stakeholders in the electricity market which can be analyzed to optimize the grid, foresee potential issues, react faster when challenges arise and build new capacities – and services – as the power landscape is changing. **[10]**

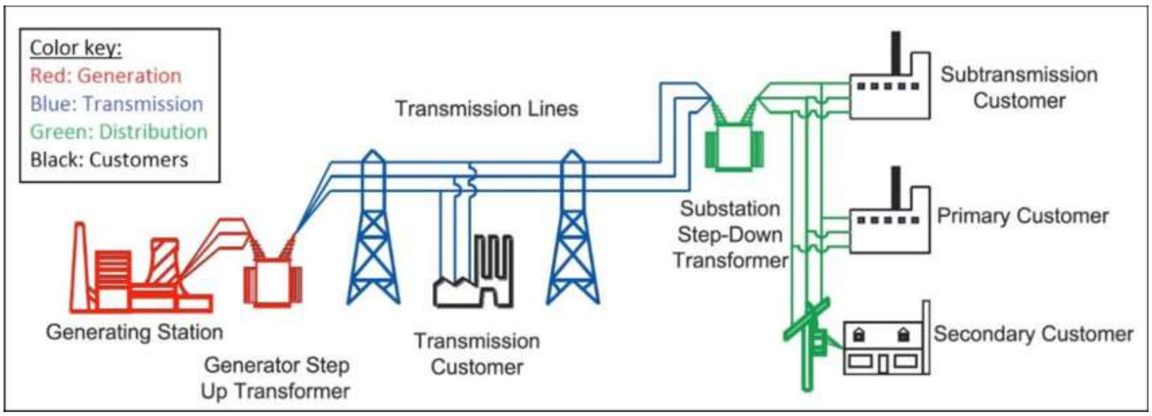


Figure 2.2: Traditional electric power system

2.2 Electrical Engineering Background

One main subject in NILM is electrical engineering, as the main task is to disaggregate the overall load profile. This section highlights its basics.
In all NILM approaches the underlying data is the measurement of voltage u and the current i. As the sensor data is a time series these two measurements can be described as:

$$u(t) = \hat{u} \cdot \sin(\omega t + \phi u)$$
$$i(t) = \hat{\imath} \cdot \sin(\omega t + \phi i)$$

With û and î representing the peak voltage and current and ϕu and ϕi the phase shift of the voltage and the phase shift of the current in regards to the start of the measurement. The overall phase shift ϕ between the voltage and the current is therefore:

$$\phi = \phi u - \phi i$$

This dependency is also highlighted in **Fig. 2.3**, where the peak values of voltage û and current î , the sinusoidal waveforms and the phase shifts of the previous equation are illustrated. Please note that the calculations are only valid for sinusoidal voltage and current waveforms.

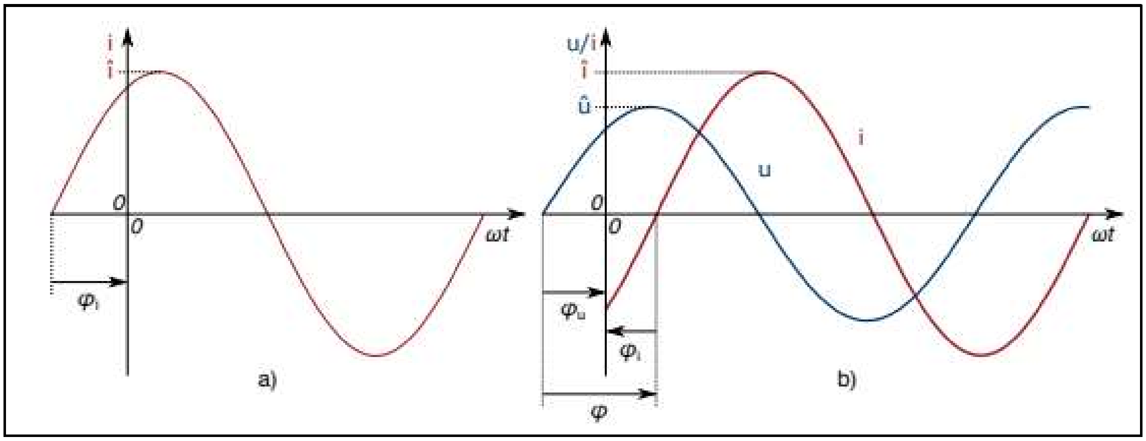


Figure 2.3: Phase shift explanation

ω = 2π/T = 2πf represents the angular frequency, with T the time period and f the frequency. For the European power network the AC mains frequency is 50 Hz. Therefore in Europe f = 50 Hz and the time period T = 20 ms.

As the mean values of the sinusoidal processes u(t) and i(t) are zero in an AC power network, the root mean square (RMS) values of these are considered:

$$U_{RMS} = \sqrt{\frac{1}{T}\int_{0}^{T} u(t)^2 dt} = \frac{\hat{u}}{\sqrt{2}}$$

$$I_{RMS} = \sqrt{\frac{1}{T}\int_{0}^{T} i(t)^2 dt} = \frac{\hat{i}}{\sqrt{2}}$$

One example is the URMS of the European power network. It is 230 V and the peak voltage value is therefore uˆ = URMS · √ 2 = 325 V . The RMS values can be seen as follows. If a resistance creates the same heat power at a DC and AC current, then the DC current value is equal to the RMS value of the AC current. This leads us to the power discussion in the upcoming section.

- **Electrical Power - Low Frequency**

Periodic electrical signals can be distinguished between sinusoidal and non-sinusoidal behavior. The electrical feature calculation of the former is a simplified case of the latter. Following equations are valid for sinusoidal electrical signals.

Active Power: Denotes the actual power, which is consumed by an electrical device. Its immediate value is called instantaneous power and can be calculated.

p(t) = u(t) · i(t) = û · î · sin(ωt + ϕ) · sin(ωt)

This formula can be rewritten as :

$$\boxed{p(t) = U_{RMS} \cdot I_{RMS} \cdot \cos(\varphi) - U_{RMS} \cdot I_{RMS} \cdot \cos(2\omega t + \varphi)}$$

Here the term URMS · IRMS · cos(ϕ) reflects the temporal mean, which is also called active power P, with the unit watt (W). It is always calculated over at least one period. The instantaneous power p oscillates around this mean with twice the frequency, which is included in the second term. Hence, active power is defined by:

$$\boxed{P = U_{RMS} \cdot I_{RMS} \cdot \cos(\varphi)}$$

Reactive Power: If there are inductances or capacitances within a power network the voltage and current waveforms are phase shifted leading to reactive power. It can be calculated by:

$$\boxed{Q = U_{RMS} \cdot I_{RMS} \cdot \sin(\varphi)}$$

In comparison to active power, reactive power cannot achieve useful work at the load. The energy is flowing back and forth from the source to the load. The unit of reactive power is volt-ampere reactive (var).

Apparent Power: The combination of active and reactive power is apparent power. It is calculated by:

$$\boxed{S = U_{RMS} \cdot I_{RMS}}$$

Apparent power is the overall consumed power and its unit is volt-ampere (VA). The dependency between active, reactive and apparent power can be visualized in following **Fig. 2.4**.

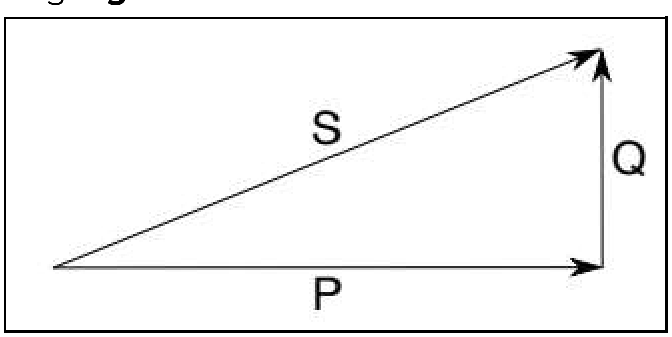


Figure 2.4: Power triangle

Therefore apparent power can also be calculated by:

$$\boxed{S = \sqrt{P^2 + Q^2}}$$

Next we look at the Fourier transform, which can be used to analyze non-sinusoidal signals.

- **Harmonics - Mid Frequency**

In a typical power network the current waveform is usually non-sinusoidal. In this case the signal is composed of several sinusoidal harmonics. The harmonic fundamental frequency equals the mains frequency, e.g. 50 Hz in Europe. Harmonic content is potentially a valuable feature in NILM and to describe it, Fourier analysis is needed which is discussed in the next subsection.

- **Fourier Analysis**

Any given periodical time series signal x(t) can be represented as a sum of sinus and cosinus functions of varying frequencies and amplitudes. This is what the theorem of Fourier states. This sum is called Fourier series and is given by:

$$\boxed{x(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty} A_n \cdot \sin(n \cdot \omega_0 \cdot t) + \sum_{n=1}^{\infty} B_n \cdot \cos(n \cdot \omega_0 \cdot t)}$$

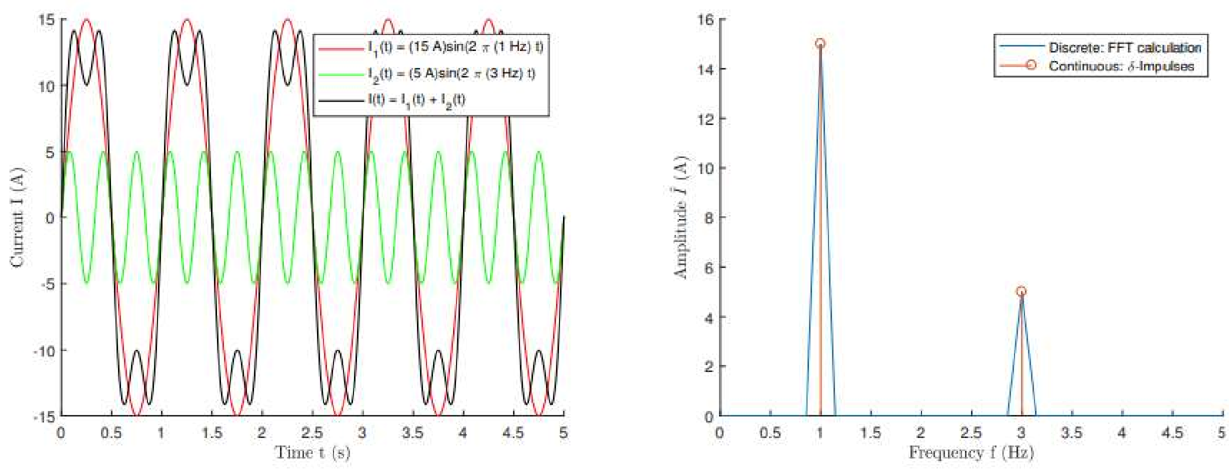
The Fourier coefficients depend on the waveform of the signal x(t) and can be calculated by:

$$A_0 = \frac{2}{T_0} \int_0^{T_0} x(t) dt$$

$$A_n = \frac{2}{T_0} \int_0^{T_0} x(t) \cdot \sin(n \cdot \omega_0 \cdot t) dt$$

$$B_n = \frac{2}{T_0} \int_0^{T_0} x(t) \cdot \cos(n \cdot \omega_0 \cdot t) dt$$

This relation is also visualized in **Fig. 2.5**. In this example the two sinus waveforms of the frequencies 1 Hz and 3 Hz lead to an aggregated and more rectangular shaped function i(t). The amplitude spectrum on the right splits the overall signal back to the two inherent frequencies.



(a) Sinus superposition (b) Amplitude spectrum

Figure 2.5: Superposition of sine waveforms and the corresponding spectrum

A signal x(t) can be further transformed into the frequency domain, containing the different sinus and cosinus frequency components. This process is called Fourier transform.

$$\underline{X}_f(f) = F\{x(t)\} = \int_{-\infty}^{+\infty} x(t) \cdot e^{-j2\pi ft} dt$$

To be able to disaggregate the overall signal in NILM it is first quantized via an analog-to-digital converter (ADC). The sensor data is therefore discrete and not continuous anymore. We discuss the discrete Fourier transform (DFT) in the next subsection as this is needed in this case.

- **Discrete Fourier Transform**

The resulting frequency spectrum is again discrete and is separated in so called frequency-bins. The distance between two bins is the resolution df of the DFT and depends on the number of samples N and the sampling period time Ts:

$$df = \frac{fs}{N} = \frac{1}{N \cdot T_S}$$

The original signal x(t) is quantized with the sampling frequency fs in Hz. To avoid aliasing effects, where the spectrum is distorted, the bandwidth B of the original signal has to be limited and the sampling theorem by Shannon has to be taken into account:

$$\boxed{f_S \geq 2 \cdot B \text{ and } f_S = \frac{1}{T_S}}$$

This means that the sampling rate has to be at least twice the bandwidth. To avoid aliasing effects far higher sampling rates are used in practical applications. The range of harmonics relevant in practice spans up to an order of about 50 or 60.

- **Power of Non-Sinusoidal Signals**

Nonlinear loads exhibit non-sinusoidal waveforms. In this case some of the above mentioned power equations are not valid anymore and alternatives are presented in this section.

To examine non-sinusoidal waveforms it is necessary to transform the original signal into the frequency space using Fourier analysis. It splits up its signal into harmonic components. As the voltage and current signals are first digitalized, the discrete Fourier transform is applied. Let In be the RMS value of the harmonic order n of the net frequency (note that the first harmonic equals the fundamental wave). Then the overall current RMS value can be calculated by:

$$\boxed{I_{RMS} = \sqrt{I_0^2 + I_1^2 + I_2^2 + \ldots}}$$

URMS is determined accordingly. Once the different harmonic voltage and current components have been calculated, the apparent power can be again calculated via:

$$\boxed{S = U_{RMS} \cdot I_{RMS}}$$

There are often non-sinusoidal currents in electrical circuits even if the mains voltage is **largely sinusoidal**. This effect occurs if nonlinear loads are attached to the power network. In this case, reactive power is composed of two different parts. Please note that the following two equations are only valid if the supply voltage is sinusoidal.

The **Displacement Reactive Power** Q1 consists of the fundamental current content and can be calculated by:

$$\boxed{Q_1 = U_{RMS} \cdot I_1 \cdot \sin(\varphi_1) \quad \textit{with } U_{RMS} \approx U_1}$$

The Deformed Reactive Power Qdf consists of the remaining harmonic content:

$$Q_{df} = U_{RMS} \cdot \sqrt{\sum_{n=2}^{\infty} I_n^2} \quad \textit{with } U_{RMS} \approx U_1$$

It is non-zero if higher harmonic current is included in the signal.
The overall **Reactive Power** can be calculated by vectorially adding these two parts:

$$Q = \sqrt{Q_1^2 + Q_{df}^2}$$

The dependencies between active power P, reactive power Q (displacement Q1 + deformed Qdf) and apparent power S are depicted in the three-dimensional **Fig. 2.6**.

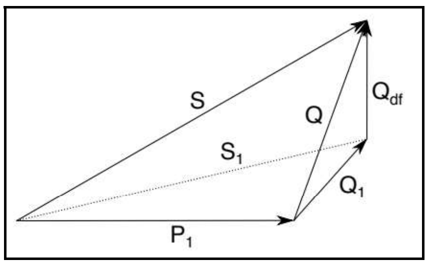


Figure 2.6: Power triangle three-dimensional

- **Line-conducted Disturbances - High Frequency**

In regards to the European directive 2004/108/EC 'electromagnetic compatibility' means "the ability of equipment to function satisfactorily in its electromagnetic environment without introducing intolerable electromagnetic disturbances to other equipment in that environment". Therefore regulations are in place which limit the emission of disturbances in the radio frequency range. Electromagnetic interference (EMI) can be divided into line-conducted disturbances and electromagnetic field-based signals. If not otherwise stated in this thesis, we refer to line-conducted electromagnetic interference signals when using the term EMI.

Even in the case that there are upper limits in place these signals are still traceable with suitable measurement hardware. As these are a potential feature in NILM we will give a short introduction in this section. Line-conducted disturbances in the radio frequency range are either introduced by galvanic, inductive or capacitive coupling or a combination of

all into the system. Galvanic coupling exists if two circuits share a common direct connection and this is the main line-conducted disturbance focus type in this thesis.
Nowadays, switched-mode power supplies (SMPS) become more and more popular because they operate more efficiently and even on a smaller size in comparison to a typical power transformer. One drawback is that they emit line-conducted disturbances. This rises on the other hand a good potential for the application in NILM, which is especially the case for the detection of electronic devices using SMPS. The relevant frequencies of such signals range up to 150 kHz and above.

After discussing the electrical engineering basics we now move on to the different electrical device types and their corresponding features which can be traced by a NILM system. Depending on the devices which should be detected a fitting sampling rate of the measurement hardware should be adjusted - taking also the Shannon theorem plus some reserves into account. This is necessary to be able to correctly calculate the above described electrical features. This should be especially considered, if smart meters are meant to be applied in NILM.**[10]**

2.3 Metering Methodologies

There're a lot of methods for electricity metering. Some of them aren't used nowadays in typical smart metering. In this section we will be exposed to the available techniques which are widely used in smart metering nowadays.

The Voltage Sensing Unit

Simple resistor dividers are widely used as voltage sensors in digital meters due to low cost. **Fig. 2.7** shows the configuration of a resistor divider type voltage sensor. The values of R1 and R2 should be chosen such that the AC mains voltage is divided down to fit the input range of the ADC of the energy measurement chip. The AC voltage is applied to R1 and output is taken from the middle point of the divider. R2 should be grounded. **[11]**

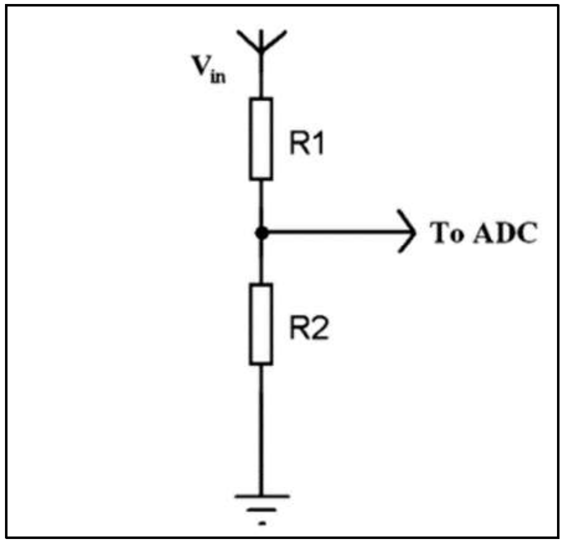


Figure 2.7: Resistor divider configuration

$$Vo = \frac{R2}{R1 + R2} Vin$$

Where,

Vo is the output voltage.

Vin is the input voltage Normally.

R1 and R2 are in the kilo-ohm scale. R1 is much greater than R2 (R1 >= 500R2). Higher values of resistors are chosen because of the lesser power dissipation.

The Current Sensing Unit

The current sensing unit typically consists of current sensors and anti-aliasing filters. Four types of current sensors are widely used in smart meters. They are:

- Hall effect-based linear current sensors.
- Current transformers.
- Shunt Resistor.
- Rogowski coils

.

(A) Hall effect-based linear current sensors

These sensors consist of a chip and a copper conduction path located near the surface of the die. The current flowing through the copper conduction path generates a magnetic field. This magnetic field is sensed by the Hall IC and converted into a proportional voltage.

(B) Current transformers

Current transformers (CTs) produce a secondary current which is proportional to the primary current. Magnetic properties of CTs are highly linear over a wide range of primary current, and temperature. The primary is connected in series with the device. The isolation is provided from primary to secondary side thus ensuring high reliability for metering devices. However, the linearity depends on the magnitude of the primary current and the impedance of the secondary. Every CT is classified according to its performance. Normally class 0.1, 0.2, 0.5 and 1 are used for metering purposes. Although CTs are more expensive than shunt resistors, they consume less power. However, CTs have nonlinear phase response at low currents and large power factors. The following Figure shows the typical through-hole type current transformers which are widely used in metering applications.

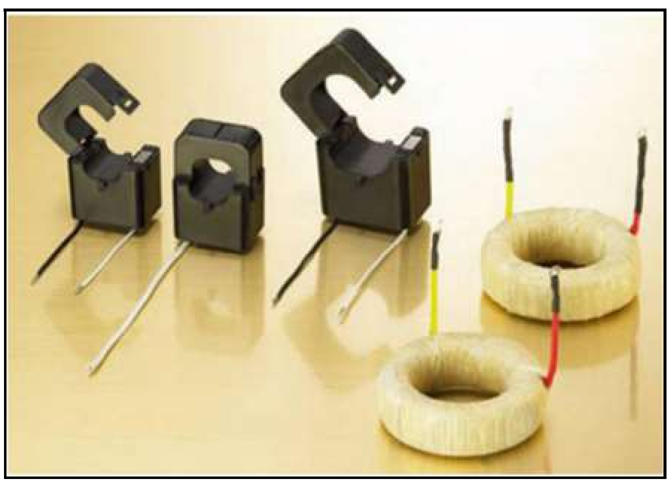


Figure 2.8: Through-hole type current transformers

(C) Shunt resistors

Shunt resistors are widely used in metering applications because of their lower cost than other types of current sensors. These sensors are simply placed in series with the load current path. Their resistances are typically in the range of (100 u ohm–500 m ohm). The power dissipated is proportional to the square of the current. Therefore, a very small resistance should be selected to minimize the heat dissipation. Shunt resistors are highly stable resistors designed with low resisting materials so that the resistance doesn't change with the current, temperature or age. The voltage across the shunt resistor is proportional to the current that flows through it. This voltage signal is fed to an energy measuring chip or to the microcontroller. Therefore, when the resistance is known, the current can be calculated according to Ohm's law. The circuit diagram of a shunt resistor is shown in **Fig. 2.9**. Although resistive shunts are inexpensive, highly linear, and immune to magnetic influences, they do not have the inherent electrical isolation, so it requires a type of isolation.

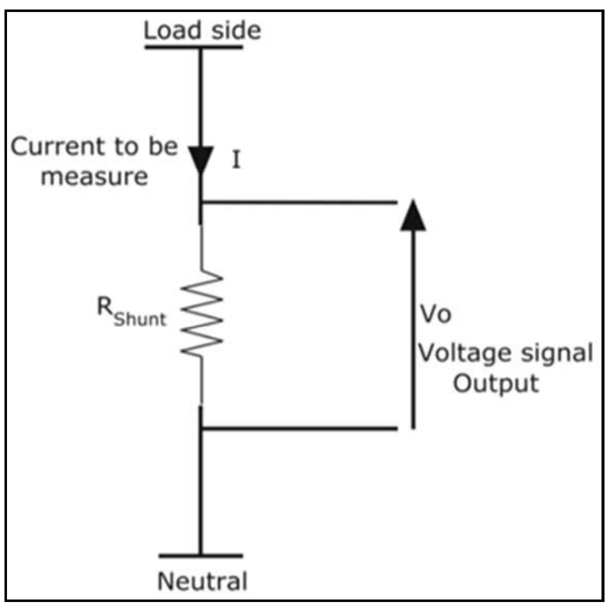


Figure 2.9: Circuit diagram of a shunt resistor

(D) Rogowski coil sensors

Rogowski coils were introduced to the electrical industry as far back as in 1912, to measure the magnetic fields, but these could not be used for current measurements, since the power produced was not sufficient to drive electromechanical equipment. With the development of solid-state electronics and microprocessor-based systems, Rogowski coils have provided a wide range of opportunities. These sensors are coils with non-magnetic core for which the name air cored is used. They give an output voltage which is proportional to the rate of change of current. They linearly convert the primary current up to all short circuit levels. Due to the absence of iron, they are saturation free. They have many advantages over conventional CTs, which include

- High measurement accuracy.
- Wide measurement range.
- Wide frequency range.
- Can withstand unlimited short circuit currents.
- Small in size and weight.
- Low production cost.

However, Rogowski coils cannot produce a voltage signal which is directly proportional to the current flow. The relationship between the output voltage and the current flow is given by

$$\boxed{V = k\frac{dI}{dt}}$$

Where:

V is the output voltage.

I is the current to be measured.

t is the time and.

k is a constant.

Current signal should be recovered from dI/dt signal. Rearranging the terms in

$$V\,dt = k\,dI$$

$$\int V dt = \int k\,dI$$

$$I = \frac{1}{k}\int V\,dt$$

According to (Previous equation), the output voltage should be integrated with respect to time to get the original current signal. Some energy measurement chips have built-in integrators to recover the current signal from Rogowski coils. Rogowski coils are commercially available in a variety of configurations, including the popular flexible type. **Fig. 2.10** shows some commercially available Rogowski coils.

Figure 2.10: Commercially available Rogowski coils

2.4 Advanced Metering Infrastructure

Advanced metering infrastructure (AMI) is a composite technology composed of several elements: consumption meters, a two-way communications channel and a data repository (meter data management). Jointly, they support all phases of the meter data life cycle — from data acquisition to final provisioning of energy consumption information to end customers (for example, for load profile presentment) or an IT application (such as revenue protection, demand response or outage management).

Advanced Metering Infrastructure (AMI) refers to systems that measure, collect, and analyze energy usage, and communicate with metering devices such as electricity meters, gas meters, heat meters, and water meters, either on request or on a schedule. These systems include hardware, software, communications, consumer energy displays and controllers, customer associated systems, meter data management software, and supplier business systems. Government agencies and utilities are turning toward advanced metering infrastructure (AMI) systems as part of larger “smart grid” initiatives. AMI extends automatic meter reading (AMR) technology by providing two way meter communications, allowing commands to be sent toward the home for multiple purposes, including time-based pricing information, demand-response actions, or remote service disconnects. Wireless technologies are critical elements of the neighborhood network, aggregating a mesh configuration of up to thousands of meters for back haul to the utility’s IT headquarters. The network between the measurement devices and business systems allows collection and distribution of information to customers, suppliers, utility companies, and service providers. This enables these businesses to participate in demand response services. Consumers can use information provided by the system to change their normal consumption patterns to take advantage of lower prices. Pricing can be used to curb growth of peak demand consumption. AMI differs from traditional automatic meter reading (AMR) in that it enables two-way communications with the meter. Systems only capable of meter readings do not qualify as AMI systems.

AMI Components

Fig. 2.11 shows the building blocks of AMI. The customer is equipped with advanced solid state, electronic meters that collect time-based data. Meters include all three types - electricity, gas, and water meters. These meters have the ability to transmit the collected data through commonly available fixed networks such as Broadband over Power Line (BPL), Power Line Communications (PLC), Fixed Radio Frequency (RF) networks, and public networks (e.g., landline, cellular, paging). The meter data are received by the AMI host system and sent to the Meter Data Management System (MDMS) that manages data storage and analysis to provide the information in useful form to the utility. AMI enables two-way communications, so communication from the utility to the meter could also take place.

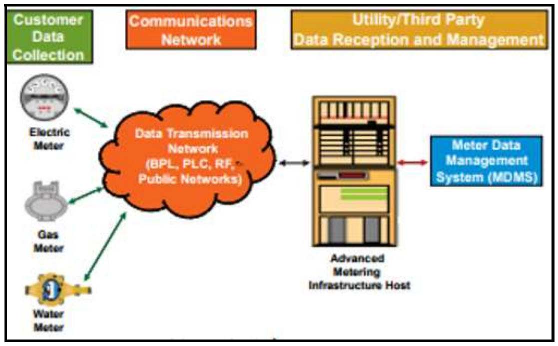


Figure 2.11: Shows the building blocks of AMI

Benefits of AMI

AMI provides benefits to consumers, utilities and society as a whole.

1. Consumer Benefits:

For the consumer, this means more choices about price and service, less intrusion and more information with which to manage consumption, cost and other decisions. It also means higher reliability, better power quality, and more prompt, more accurate billing . In addition, AMI will help keep down utility costs, and therefore electricity prices. And, as members of society, consumers also reap all the benefits that accrue to society in general, as described below.

2. Utility Benefits:

Utility benefits fall into two major categories, billing and operations. AMI helps the utility avoid estimated readings, provide accurate and timely bills, operate more efficiently and reliably, and offer significantly better consumer service. AMI eliminates the vehicle, training, health insurance, and other overhead expenses of manual meter reading, while the shorter read-to-pay time advances the utility's cash flow, creating a one-time benefit. And consumer concerns about meter readers on their premises are eliminated. Operationally, with AMI the utility knows immediately when and where an outage occurs so it can dispatch repair crews in a more timely and efficient way. Meter-level outage and restoration information accelerates the outage restoration process, which includes notifying consumers about when power is likely to return. Using AMI, the utility can receive significant benefits from being able to manage customer accounts more promptly and efficiently, starting with the ability to remotely connect and disconnect service without having to send personnel to the customer site. Similarly, many maintenance and customer service issues can be resolved more quickly and cost-effectively through the use of remote diagnostics. AMI enables new programs and methods for creating and recovering revenue such as distributed generation and prepayment programs. AMI also provides vast amounts of energy usage and grid status information that can be used by consumers to make more informed consumption decisions and by utilities to make better decisions about system improvements and service offerings. Instead of relying on rough estimates, engineers armed with AMI's detailed knowledge of distribution loads and electrical quality can accurately size equipment and protection devices, and better understand distribution system behavior. This huge increase in valuable information helps the utility:

- Assess equipment health
- Maximize asset utilization and life
- Optimize maintenance, capital and O&M spending
- Pinpoint grid problems
- Improve grid planning
- Locate/ identify power quality issues
- Detect/reduce energy theft

3. Societal Benefits:

Society, in general, benefits from AMI in many ways. One way is through improved efficiency in energy delivery and use, producing a favorable environmental impact. It can accelerate the use of distributed generation, which can in turn encourage the use of green energy sources. And it is likely that emissions trading will be enabled by AMI's detailed measurement and recording capabilities. A major benefit of AMI is its facilitation of demand response and innovative energy tariffs. During periods of high energy demand, a small reduction in demand produces a relatively large reduction in the market price of electricity. And reduced demand can avoid rolling blackouts. According to Edison Electric Institute (EEI), the direct costs (e.g. power costs) of rolling blackouts in California have been estimated at tens of millions of dollars. Business and consumer losses may be many times higher. Hence, a modest demand response capability could produce a societal benefit worth billions of dollars. The benefits accrued may vary depending on the type of demand response programs initiated. For instance, demand response distributed to the individual premise in forms like thermostat and pool pump control allows load to be reduced without sacrificing consumer satisfaction. However, even just shifting demand away from peak hours through time-of-use tariffs can have major benefits, including the reduced cost to both utilities and consumers by deferring building new, expensive peak generation facilities. There is also a societal fairness issue that AMI addresses. Full deployment of AMI results in the elimination of old and obsolete electromechanical meters that tend to slow down as they age. Modern AMI meters maintain their accuracy over time, resulting in a more equitable situation for all consumers. In addition, modern meters are self monitoring, making it easier to identify inaccurate measurements, incorrect installations and, especially, electric energy theft.

4. Added Benefits:

AMI serves as a Modern Grid Platform: going beyond AMI to achieve a truly modern grid produces additional large improvements in the operations of an electric utility. The list of benefits includes:

- Greatly improved outage management system (through links with GIS and real time consumer status)
- Improved system planning process and results
- Improved distribution asset management programs including equipment health assessment and condition-based maintenance
- Advanced distribution management systems (distribution automation, integrated operation of DR (and DER), micro-grid operation, self-healing,etc.)
- Improved mobile workforce management and operations
- New opportunities for consumer choice and new retail services
- Improvements to power quality issues
- Reduced environmental impact
- Distribution system support of transmission operations (transmission congestion relief, voltage support, loss reduction)

Challenges in AMI Implementations

What are Challenges in an AMI Implementation?

AMI implementations are massive transformation programs which typically require a timeframe of between 3-5 years for a full-scale rollout depending on the customer base of the utility. The following are key challenges in any AMI implementation:

- Cost: Deploying smart meters requires huge capital investment and hence it is crucial that the utility company is confident of successful implementation.

- Data Privacy: There are many concerns related to privacy of consumption data being raised as Smart Meters are installed at more and more locations. The meters' data can be mined to reveal details about customers' habits like when they eat, how much television they watch and what time they go to sleep. The retention and storage of this data make it vulnerable to security breaches as well as government access.

- Technology Transformation: The imminent rollout of smart metering in the short to mid-term future will require vital investments in information technology for setting up turnkey projects for smart metering network and application infrastructure. It will be necessary to build up components like:

    - Mass deployment of smart meters
    - Meter reading databases and applications
    - Data warehouses and analytic tools & data migration programs
    - Customer interfacing applications and portals
    - Integration with existing systems

These components and underlying infrastructure will be critical success factors for generating maximum success from smart meter rollout programs. AMI implementation brings with it a transformation in the entire utility business process, and impacts the entire span of utility operation & stakeholders. Hence validation of AMI implementation becomes a very crucial task. It becomes imperative that the utility validation partner in these kinds of programs understands the business domain, and ensures that the application not just conforms to the business requirements but is also able to meet other criteria (e.g., performance, data quality & security, application security).

2.5 Electricity Theft and Tampering

Electricity theft, defined here as “intentional theft or nonpayment for electrical services by way of a variety of technical or non-technical means,” is currently an important economic and technical issue. Economically, worldwide electricity theft losses were most recently (2014) estimated to be $89.3 billion. Technically, various approaches have been developed to detect and mitigate against theft.

The smart grid, one advanced energy technologies of interest herein, involves the widespread adoption of digital meters, sensors and controllers to provide improved quality and resiliency of the electric grid by remote monitoring and control of all aspects of electricity transmission and distribution (T&D). As a byproduct of the smart grid is that electricity theft can be mitigated to some degree by improved monitoring, improved meters, and improved record keeping. Electricity theft is considered as the intentional nonpayment for electrical services by way of a variety of technical or nontechnical means. Consistent with the authors contend that E-theft is of three types:

1. **Outright Theft, which is accomplished by:**
- Tapping an overhead line to create a new, illegal connection.
- Induction Coupling whereby energy from a power line is collected by electromagnetic induction without physically connecting to the line.
2. **Fraud, which is accomplished by**
- Bypassing a meter to prevent it from measuring the power consumed.
- tampering with a meter to cause it to output a more favorable reading for the customer. This is subdivided into mechanical and digital/smart meter methods.
3. **Billing Issues**
- Deliberate non-payment of bills.
- Billing irregularities, both intentional (bribing officials to ignore use) and unintentional (poor record keeping practices).
- Although only one method is termed as explicitly theft, all of these issues involve consuming electricity which is not paid for.

1. **Outright Theft Outright theft**

Involves a one stealing electricity when they are not already a known customer to a utility. Here an electricity theft aims to create a new, and illegal, connection to the T&D system without the approval of the owner of the T&D system.

a) Tapping

Tapping, as seen in **Fig. 2.12** , involves creating an unauthorized connection to overhead or buried lines on the line side of the distribution transformer [33]. In operation, tapping can be used to connect a premise, or equipment, to the electric grid where no prior connection to the grid existed. As a method of E-theft, tapping has been in use in the United States since at least the 1890’s .

Figure 2.12: Contemporary Example of Tapping in America, using jumper cables to connect directly to the overhead lines

Tapping offers a significant amount of risk of electrocution and subsequent death from this form of electricity theft to both a perpetrator and to innocent bystanders. Additionally, damage to T&D equipment, including transformers, is also possible if the thief shorts a connection. Despite the inherent danger of this technique, it has been adapted and refined for field-use by US military engineers in a deployable system that can be thrown up to a power line to charge equipment.**[13]**

b) Induction Coupling

One method that appears in the public conscious is that of placing a large coil under a high voltage power line in order to steal electricity. Such an approach is inherently a form of tapping by induction coupling. However, aside from a few anecdotal reports of electricity theft by induction coupling, issues exist with such schemes. Primarily, it is generally improbable to make a return on the investment due to the required large amount of copper needed to create a sufficiently large coil . Although the actuality of induction coupling electricity theft is low, there are instances of induction power light bulbs as seen in artistic installations where fluorescent lights are lit by induction .

2. Fraud

Fraud in electricity theft involves performing an action to record less consumption on an electricity bill or tampering with metering equipment, which is owned by the power utility, to make it record less (or no) consumption than what was actually consumed.

a) Bypassing Existing Meetings

Bypassing a meter extends the concept of tapping, but here a prior electric is accomplished by directly connecting the house wiring to the wires coming into the meter wiring around the meter see **Fig. 2.13** for two examples of accomplishing this through the use of automotive jumper cables and screwdrivers). This form of theft can either completely disconnect the meter from the system or leave the meter connected in addition to the bypass so that the meter records some amount of usage, though less than before . An additional variation on bypassing includes using a second (spare) meter for part of the billing period to avoid all usage being recorded. During whatever process a thief might use to bypass a meter, it would be counterproductive for a thief to notify a utility to shut off power to a premise and it is assumed that all connections would therefore be made to live wires. Therefore, this method of theft also offers a significant amount of personal risk.

Figure 2.13: Two Examples of Bypassing a Meter Using Automobile Jumper Cables and Screwdrivers

a) Meter Tampering

Tampering with an electric meter has been a concern for over one hundred years and it is known that designing meters to avoid tampering was a primary consideration by the late 1890's, and is still a contemporary concern.

i) Mechanical Meter

Tampering Methods of tampering with a meter and protection against tampering may have evolved over time, but where mechanical meters are in use some methods have changed little in over one hundred years. For example, the following methods have been viable for mechanical meter tampering from the 1890s to contemporary times:

- Mechanically stopping the disc from moving, thus stopping it from recording the amount of energy used
- Tampering with calibration screws.
- Placing magnets around the meter to disrupt the meter's operation · inserting foreign matter, such as honey, glue, insects (such as ants and spiders), iron filings or grease to slow the movement
- Contaminating the meter bearing
- Disconnecting the neutral conductor of a three-conductor meter.
- Mechanical meters do not register power consumption under this condition
- All safety aspects of having a neutral conductor are also lost damaging the movement so that it fails to operate properly.
- Reserved current tampering: physically turning the meter upside-down (inverting) causing the meter to run backwards ·
- Placing the meter out of plumb ·
- Altering the internal wiring of meters .

The locations of where some of these methods are applicable are highlighted in **Fig 2.14**. It is important to note that many of these methods involve the thief being in contact with the internal wiring of the meter or internal components. Therefore, this method of theft may provide no more safety than either bypassing or tapping.

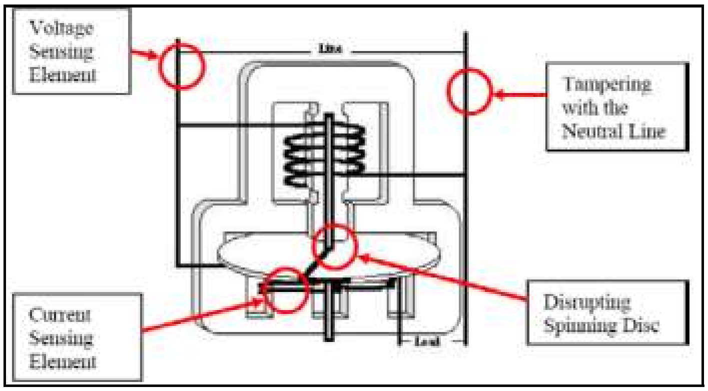


Figure 2.14: Single-Phase Mechanical Watt-hour Meter with Typical Locations Susceptible to Tampering Highlighted

ii) Digital Meter Tampering

Digital meters, considered herein as any smart, digital or electronic meter, have been presented as a solution to many problems in metering, such as manpower reductions, revenue protection , and economics. However, digital meters are not immune to electricity theft and introduce other tangential issues to electricity theft considerations, such as data and consumer privacy , and big data problems of data storage and data mining. Additionally, the manpower reduction benefit can potentially facilitate theft by losing the frequent personal contact of meter readers . In general there are two types of digital meters :

1. **hybrid digital-mechanical meters:**

Which employ a digital counter, but otherwise measure usage mechanically

2. **wholly electronic meters:**

Digital meters generally operate as presented in **Fig. 2.15** , where current and voltage sensors monitor power flow into a premise, analogue-to-digital (ADC) devices convert sensor readings to values for a digital signal processor (DSP), with the result displayed visually and possibly transmitted to the utility via an automatic meter reading (AMR) unit.

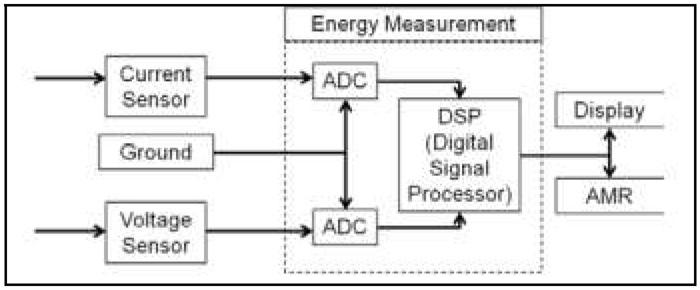


Figure 2.15: Conceptualization of Single-Phase Digital Watt Hour Meter .

As we know digital meters have some general advantages over mechanical meters, a digital meter typically has no disc to mechanically stop and few have moving parts that can be interfered. Additionally, some digital meters have tilt and magnetic interference sensors to record when a meter was tampered with, possibly transmitting this data to the utility.

In summary, digital meters have the following benefits:

- Greater accuracy in some situations.
- Elimination of moving parts, thereby removing the viability of some electricity theft methods.
- Real-time communication and monitoring of usage by utilities·
- AMRs record more detailed usage data and can transmit such data to the utility.

  However, digital meters also have multiple vulnerabilities relative to electricity theft, including·
- Potential hacking concerns
- Disconnecting the neutral wire prevents some digital meters from recording usage.

Therefore, as with mechanical meters, a variety of issues and benefits exist with digital meters. Although the impact of digital meter tampering is unknown, Rengarajan and Loganathan. posited that it was less than tapping because electronic meters would transmit a warning once a seal was breached. However, this relies on the assumption that a thief cannot avoid tripping that sensor, as mentioned above there are multiple methods to hack a digital meter.

2.6 Programing Languages

This project needed a lot of technical background including theoretical knowledge, practical knowledge, and we had to use a lot of different programming languages and tools.

C programming Language

C is a general-purpose, procedural computer programming language supporting structured programming, lexical variable scope, and recursion, with a static type of system. By design, C provides constructs that map efficiently to typical machine instructions.

C++ programming language

C++ is a powerful general-purpose programming language. It can be used to develop operating systems, browsers, games, and so on. C++ supports different ways of programming like procedural, object-oriented, functional, and so on. This makes C++ powerful as well as flexible.

Python programming Language

Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation.

Java programming Language

Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible

Dart programming language

Dart is an open-source general-purpose programming language. It was originally developed by Google and later approved as a standard by ECMA. Dart is a new programming language meant for the server as well as the browser. Introduced by Google.

2.7 Communication Protocols

- **Communication Protocols:**

They are a set of rules that allow two or more communication systems to communicate data via any physical medium. The rules, regulations, synchronization between communication systems, syntax to be followed and semantics are all defined by the term protocol. Protocols can be implemented by both hardware and software or combination of both.

In our project we used :-

1- UART : stands for Universal Asynchronous Receiver Transmitter which is a hardware communication protocol that uses asynchronous serial communication with configurable speed. Asynchronous means there is no clock signal to synchronize the output bits from the transmitting device going to the receiving end. UART is used to communicate with the modem to transmit and receive data or commands to the modem.
UART embeds start and stop bits with actual data bits, which defines the start and end of the data packet.

When the receiver end detects the start bit, it starts to read the data bits at specific baud rate meaning both transmitting and receiving peripherals should work under the same baud rate. UART works under half duplex communication mode meaning it either transmits or receives at a time.

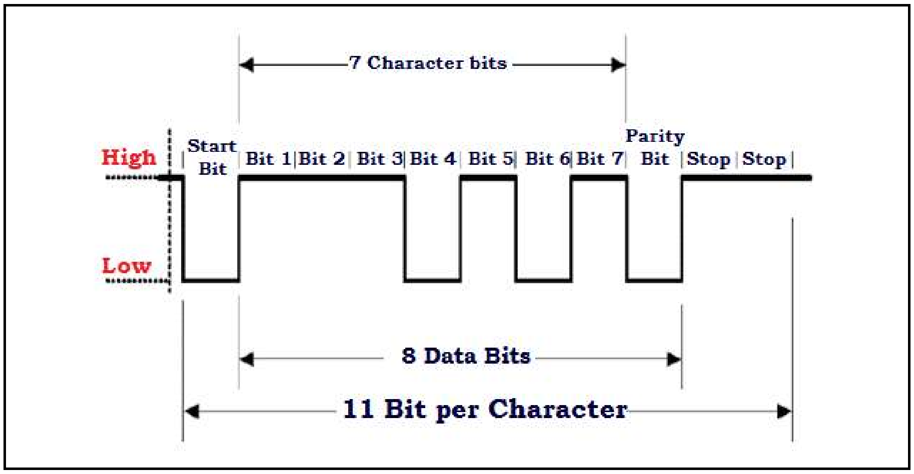


Figure 2.16 : UART Frame

2- SPI : stands for Serial Peripheral Interface which is a synchronous serial communication interface specification used for short-distance communication, primarily in embedded systems. It's featured by the high speed which is in MegaByte and also 100% throughput. SPI is used to communicate with the Measurement unit and Display unit.

SPI is a master to slave communication protocol. In SPI, the master device first configures the clock at a particular frequency. Furthermore the SS line is used to select the appropriate slave by pulling the SS line low where it is normally held high.

The communication is established between the selected slave and the master device as soon as appropriate slave device is selected.

SPI is a full duplex communication protocol. SPI doesn't limit data transfer to 8 bit words.

- **Advantages of SPI Communication Protocols**

The advantages of SPI Communication Protocol are as follows:

- Faster than asynchronous serial communication protocol.
- Support multiple slaves connectivity.
- Universally accepted protocol and low cost.

- Disadvantages of SPI Communication Protocol

The disadvantages of SPI Communication Protocol are as follows:

- Requires more wires than other communication protocols.
- Master device should control all slave communications (slave-slave communication is impossible).
- Numerous slave devices leads to circuit complexity.

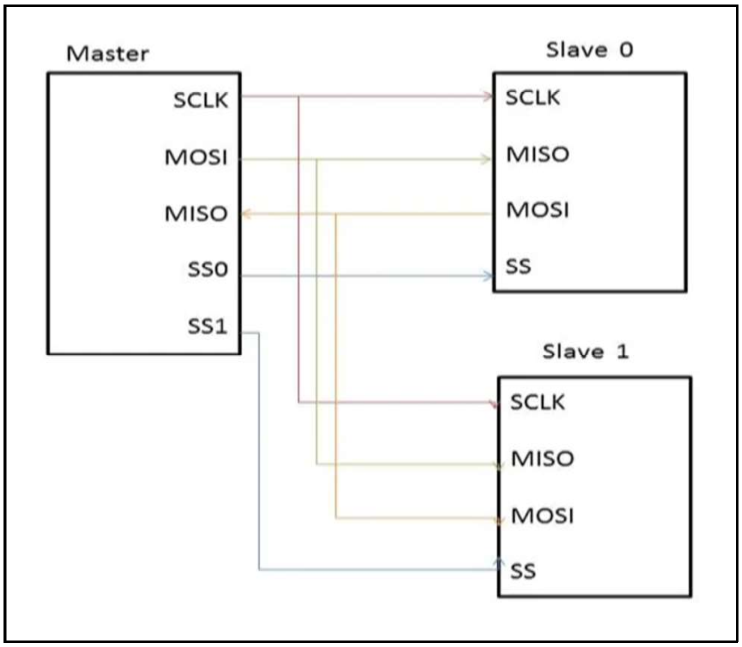


Fig 2.17: SPI Communication Protocol

3- I2C: Inter Integrated Circuit (I2C) is a serial communication protocol developed by Philips Semiconductors. The main purpose of this protocol is to provide easiness to connect peripheral chips with the microcontroller. In embedded systems, all peripheral devices are connected as memory mapped devices to the microcontroller.

I2C necessitates two wires SDA (Serial Data Line) and SCL (Serial Clock Line) to carry information between devices. These two active wires are said to be bidirectional.

I2C protocol is a master to slave communication protocol. Each slave is been provided with unique address. In order to establish communication, master device initially sends the target slave address along with R/W (Read/Write) flag. The corresponding slave device will move into active mode leaving other devices in off state.

Once the slave device is ready, communication starts between master and slave devices. One bit acknowledgment is replied by the receiver if the transmitter transmits 1 byte (8 bits) of data. A stop condition is issued at the end of communication between devices.

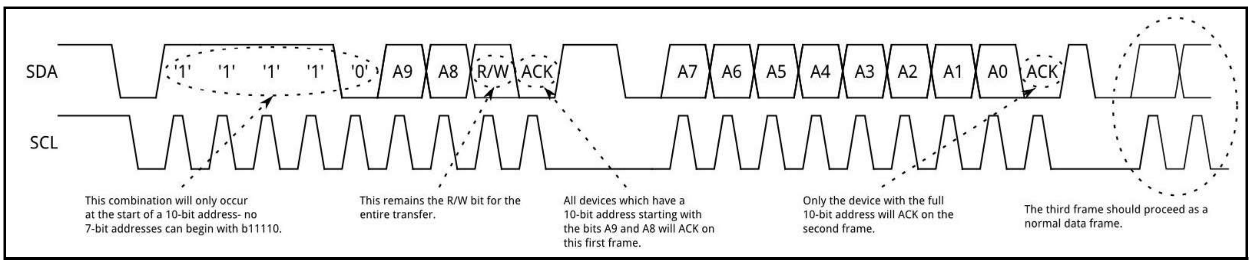


Fig 2.18 : I2C Communication Protocol

- Advantages of I2C Communication Protocols

- Provides good communication between onboard devices which are accessed infrequently.
- Addressing mechanism eases master slave communication.
- Cost and circuit complexity does not end up on the number of devices.

- Disadvantages of I2C Communication Protocols

The biggest disadvantage of I2C Communication Protocols is its limited speed.

2.8 Linux OS

What is Linux?

Linux is the name given to the kernel and a series of Unix operating systems under GNU GPL. In many respects, Linux is similar to other operating systems such as Windows and macOS. Like them, Linux can have a graphical interface and the same types of desktop software that you are used to, such as word processors, photo editors, video editors, etc. In 1991 Linus Torvalds decided to create a new free operating system kernel as a personal project. That results in creating one of the greatest software programs in history, starting from a few files of C code to 23.3 million lines of source code. You are probably already using Linux, even if you are not aware of it since it is present in the software of a large number of devices that we use on a daily basis. Even most of the Internet web pages that we visit have probably been generated by Linux servers. Similarly, most companies and individuals choose Linux for their servers because it is secure, flexible, and can receive excellent support from a large community of users.

What're the Features of using operating systems?

Any OS should provide the following functionalities:

- Multi-Tasking & Scheduling

  Even if you have only a single core processor, you are able to run multiple processes at the same time because of the scheduling and the process management which are provided by the OS.

- Hardware Management

  You don't need to write your own C code or assembly to make the hardware work.

- Frameworks Supporting

  When you are dealing with a microcontroller, you may write C/C++ code or even assembly to achieve the functionalities, but when you are dealing with an OS there's a lot of freedom to use any other high-level languages such as JAVA, Python, etc.

What're the Features of Linux?

Beside the previously mentioned functionalities, Linux provides moreover than that. Linux is featured by:

- Popular OS and community support.
- Providing code portability because of the POSIX interface.

POSIX stands for Portable Operating System Interface. It's a family of standards specified by IEEE for maintaining compatibility among operating systems. Therefore, any software that conforms to POSIX standards should be compatible with other operating systems that adhere to the POSIX standards. So, when you write code over Linux it can be ported on any other hardware that can run Linux.

- Not only open-source, it's also free.

That means: you can modify the source code **[14]** as it's available for free; you can redistribute it and you can use it in commercial projects.

Embedded Linux

An embedded system is a set of computer hardware and software based on a microcontroller or microprocessor, controlled by a real-time operating system or RTOS, limited memory, and that can vary both in size and complexity. Embedded Linux is a type of Linux operating system/kernel that was designed to be installed and used in embedded devices or systems.

Although it uses the same kernel, embedded Linux is a little bit different from the standard operating system. First of all, it is customized for embedded systems and, therefore, is much smaller in size, requires less processing power, and has minimal features. The Linux kernel is modified and optimized as an embedded Linux version. Such a Linux instance can only run applications created specifically for the device.

Embedded Linux is flexible, low-cost, open-source, and gets adapted to specific-purpose microprocessors. Compared to proprietary embedded operating systems, Linux allows multiple software, development, and support vendors; it has a stable kernel and provides the ability to read, modify, and redistribute source code. It also allows for a highly modular building block approach to building a custom system, leading to greater flexibility. Also, Linux is one of the most popular Operating system which is used in IoT devices this from the IoT Developer Survey 2018 conducted by Eclipse Foundation, it was found that 71.8% of the respondents like or use Linux-based OS as shown in **Fig 2.19**.**[15]**

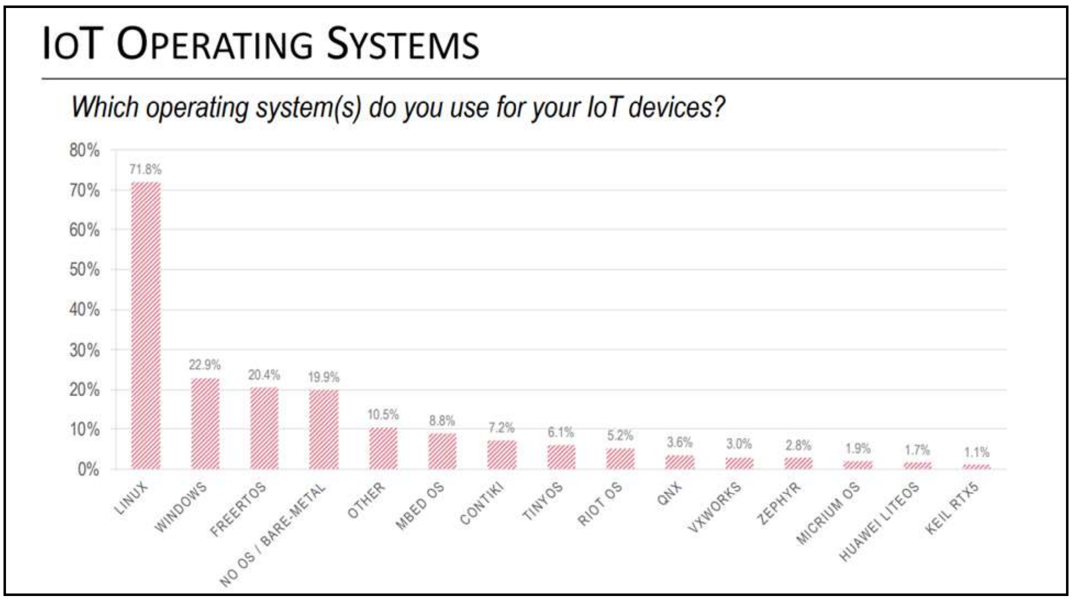


Figure 2.19: Popular IoT operating systems in early 2018. Source: Brown 2018.

Process vs Daemon

A program can be run in different modes depending on the operating system. These modes are known as Processes, Daemons.

Process is a program or command that is actually running on the computer as shown in **Fig. 2.20**.

Daemon is a process which runs in the background and is not interactive. It has no controlling terminal on its own from the user's perspective from the desktop as shown in **Fig. 2.21**.

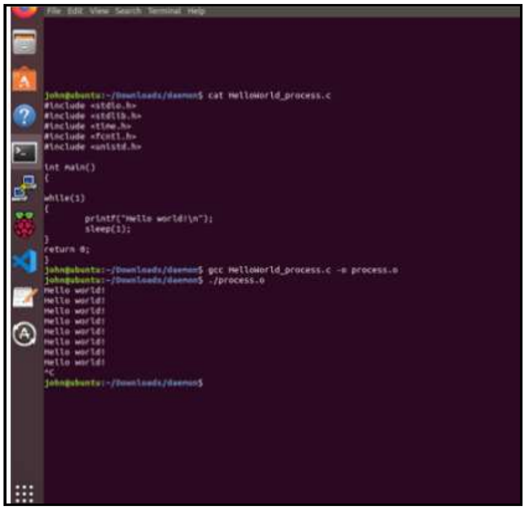


Figure 2.20 : Process example **Figure 2.21 : Daemon example**

Approaches to Inter-process Communication

The different approaches to implement interprocess communication are given as follows

Pipe:

A pipe is a data channel that is unidirectional. Two pipes can be used to create a two-way data channel between two processes. This uses standard input and output methods. Pipes are used in all POSIX systems as well as Windows operating systems.

Socket:

The socket is the endpoint for sending or receiving data in a network. This is true for data sent between processes on the same computer or data sent between different computers on the same network. Most of the operating systems use sockets for inter-process communication.

Signal:

Signals are useful in inter-process communication in a limited way. They are system messages that are sent from one process to another. Normally, signals are not used to transfer data but are used for remote commands between processes.

Shared Memory:

Shared memory is the memory that can be simultaneously accessed by multiple processes, so that the processes can communicate with each other. All POSIX systems, as well as Windows operating systems use shared memory.

Message Queue:

Multiple processes can read and write data to the message queue without being connected to each other. Messages are stored in the queue until their recipient retrieves them. Message queues are quite useful for inter-process communication and they are used by most operating systems.

A diagram that demonstrates message queue and shared memory methods of inter-processes communication as follows

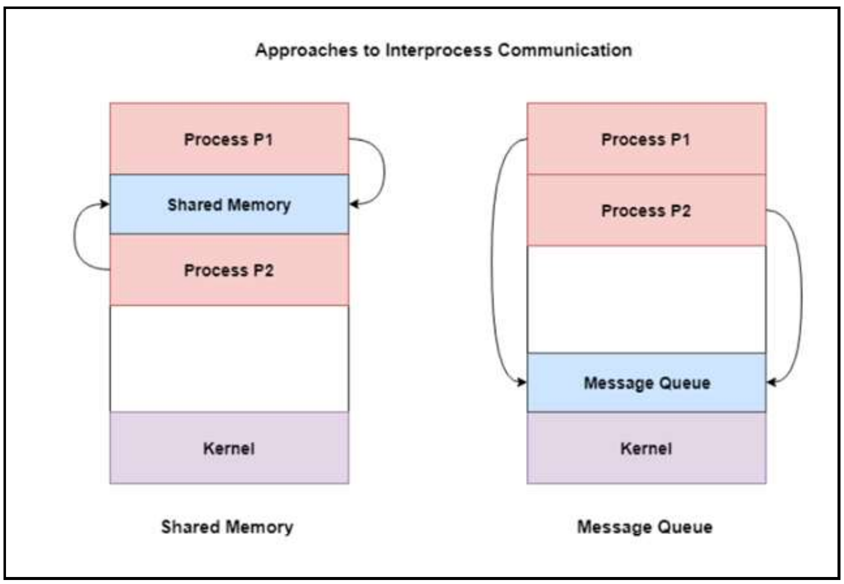


Figure 2.22 : Shared memory vs Message queue

2.9 Software Tools

- In Machine Learning:

1. Anaconda:

Anaconda offers the easiest way to perform Python/R data science and machine learning on a single machine. Start working with thousands of open-source packages and libraries today. Anaconda was built by data scientists, for data scientists. More than 20 million people use our technology to solve the toughest problems.

Figure 2.23: Anaconda logo

2. TensorFlow Framework:

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications. TensorFlow makes it easy to build and deploy ML models. TensorFlow offers multiple levels of abstraction so you can choose the right one for your needs. Build and train models by using the high-level Keras API, which makes getting started with TensorFlow and machine learning easy. If you need more flexibility, eager execution allows for immediate iteration and intuitive debugging. For large ML training tasks, use the Distribution Strategy API for distributed training on different hardware configurations without changing the model definition.

Figure 2.24: TensorFlow Framework logo

3. Google Colab:

Colaboratory is a data analysis tool that combines text, code, and code outputs into a single document. Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML, LaTeX and more. When you create your own Colab notebooks, they are stored in your Google Drive account. You can easily share your Colab notebooks with co-workers or friends, allowing them to comment on your notebooks or even edit them.

Figure 2.25: Google Colab logo

4. Jupyter Notebook:

JupyterLab is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality. The Jupyter Notebook is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience.

Figure 2.26: Jupyter Notebook logo

- In Frontend development:

1. Visual Studio code:

A standalone source code editor that runs on Windows, macOS, and Linux. The top pick for JavaScript and web developers, with extensions to support just about any programming language.. The best comprehensive IDE for .NET and C++ developers on Windows. Fully packed with a sweet array of tools and features to elevate and enhance every stage of software development. Go beyond syntax highlighting and autocomplete with IntelliSense, which provides smart completions based on variable types, function definitions, and imported modules.

Figure 2.27: Visual studio code logo

- In Mobile Development :

1. Flutlab:

FlutLab is a modern Flutter online IDE and the best place to create, debug, and build cross-platform projects. FlutLab is a web portal for modern software developers that completely replace traditional local and desktop tools. People can study and work from anywhere, add teammates to the projects, share or sell their code with one click using our online IDE and ecosystem built around it.

It's a web site that deals with flutter and enables us to create our mobile application on. It provides us with the same capabilities as vs code and android studio like testing your application on an emulator, building a web application and creating an APK. But it’s better than them in some points like:

- Providing us with a less size application.
- We use it online, so if anything happens to your computer you don’t have to worry about your files.
- If your computer is low quality, it’s better for you to use it because it’s quicker than vs code and android studio in building projects.

In fact, it also has some disadvantages like:

- You have the right to create only two projects on your account.
- If you don't make edits to your project for a while, it will expire.
- You can’t debug your code.

Figure 2.28: Flutlab logo

- In Backend Development:

1. Microsoft SQL database

SQL was firstly developed by IBM for querying and altering relational databases, using declarative statements.SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization (ISO) in 1987. Although it is often called a “query language”, SQL is much more than that. SQL can define data structure, modify data in a database, specify security constraints, and can perform many more tasks such as:

- SQL can retrieve data from a database
- SQL can insert records in a database

- SQL can update records in a database
- SQL can delete records from a database
- SQL can create new databases
- SQL can create new tables in a database
- SQL can create stored procedures in a database
- SQL can create views in a database
- SQL can set permissions on tables, procedures, and views
- SQL can execute queries against a database

Figure 2.29: Microsoft SQL

2. Spring Boot

Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run". We take an opinionated view of the Spring platform and third-party libraries so you can get started with minimum fuss. Most Spring Boot applications need minimal Spring configuration. If you're looking for information about a specific version, or instructions about how to upgrade from an earlier release, check out the project release notes section on our wiki.

3. Spring MVC

A Spring MVC is a Java framework which is used to build web applications. It follows the Model-View-Controller design pattern. It implements all the basic features of a core spring framework like Inversion of Control, Dependency Injection.

A Spring MVC provides an elegant solution to use MVC in spring framework by the help of DispatcherServlet. Here, DispatcherServlet is a class that receives the incoming request and maps it to the right resource such as controllers, models, and views.

4. Spring Security

Spring Security is a powerful and highly customizable authentication and access-control framework. It is the de-facto standard for securing Spring-based applications.

Spring Security is a framework that focuses on providing both authentication and authorization to Java applications. Like all Spring projects, the real power of Spring Security is found in how easily it can be extended to meet custom requirements.

<u>Features</u>

- Comprehensive and extensible support for both Authentication and Authorization
- Protection against attacks like session fixation, clickjacking, cross site request forgery, etc
- Servlet API integration
- Optional integration with Spring Web MVC

5. Stripe payment

It is processing software and application programming interfaces for e-commerce websites and mobile applications.

6. Postman

It is a helpful tool for testing and exploring APIs. And that's great! It's a large part of what the tool was created to do. But the utility actually goes quite a bit deeper than that. This workspace attempts to introduce folks to some not-quite-beginner level features of Postman that are either fun, useful, or perhaps both.

Application Programming Interface or API is no longer an unpopular term in software development. In a three-tier architecture, APIs lay in the domain or business logic tier and ensure the continuity of data transformation between the two other components.

The significance of APIs is reflected in how they help developers save a vast amount of resources when developing applications. With over thousands of public APIs available, developers can access and incorporate almost every shared function into their software without having to start from scratch.

Figure 2.30: Postman

7. Thymeleaf

It is a modern server-side Java template engine for both web and standalone environments. Thymeleaf's main goal is to bring elegant natural templates to your development workflow HTML that can be correctly displayed in browsers and also work as static prototypes, allowing for stronger collaboration in development teams. With modules for Spring Framework, a host of integrations with your favourite tools, and the ability to plug in your own functionality, Thymeleaf is ideal for modern-day HTML5 JVM web development although there is much more it can do.

Figure 2.31: Thymeleaf

- In Embedded Linux:

1. Altium Designer

Altium offers a unified design environment, empowering engineers with a single view of every aspect of the PCB design process from schematic, to PCB layout, to design documentation. By accessing all design tools in one place, engineers can complete their entire design process within the same intuitive environment and deliver high-quality products quickly. Altium Designer combines a multitude of features and functionality, including:

- Advanced routing technology
- Support for cutting-edge rigid-flex board design
- Powerful data management tools
- Powerful design reuse tools
- Real-time cost estimation and tracking
- Dynamic supply chain intelligence
- Native 3D visualizations and clearance checking
- Flexible release management tools
- It provides a 6 months free trial for students which is a great opportunity for us.

Figure 2.32: Altium Designer

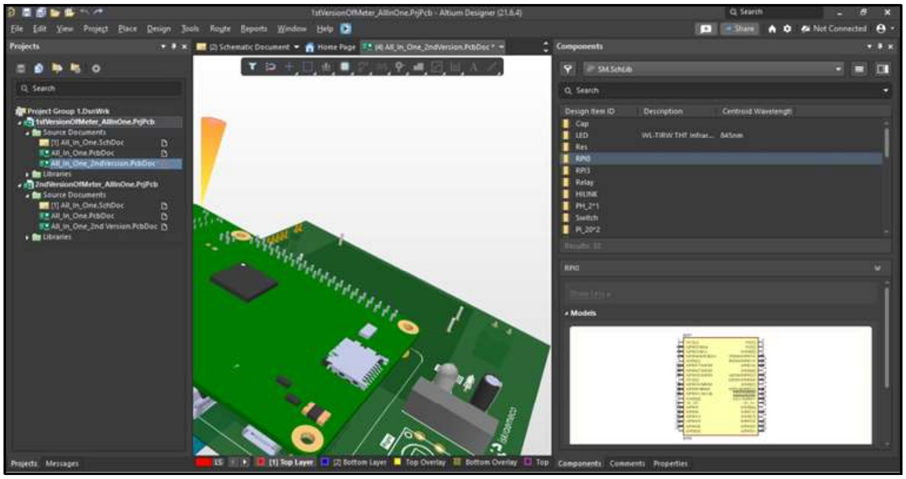


Figure 2.33: Altium Designer project

2. Buildroot

It is a simple, efficient and easy-to-use tool to generate embedded Linux systems through cross-compilation. It has a simple structure that makes it easy to understand and extend. It relies only on the well-known Makefile language. It is also an open-source tool.

Figure 2.34: Buildroot

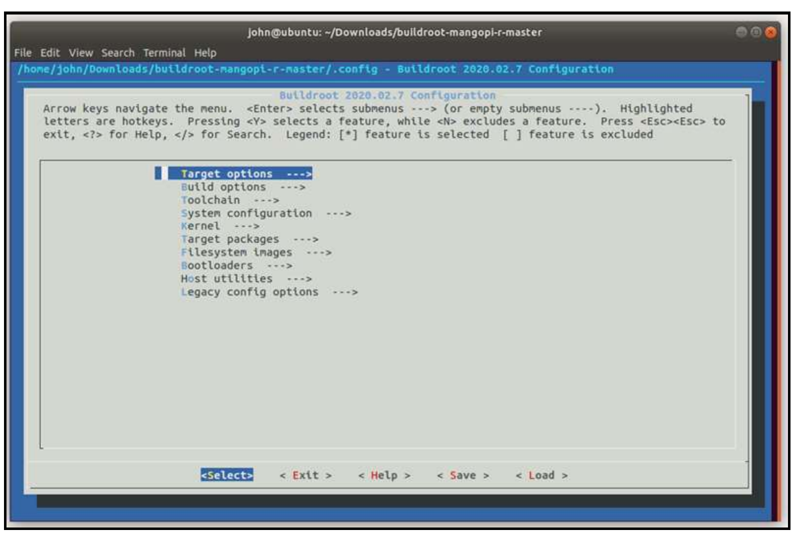


Figure 2.35: Buildroot menuconfig

2.10 Hardware Components

In our project, we used:

- Various ARM-based microcontrollers that control our system such as, STPM33, and Raspberry Pi 0 .
- F1c200s Soc
- Limit Switch
- TFT display
- Push button switches
- Hilink Power Supply

Those Choices was based on a study of the hardware criteria to make sure Each unit of hardware can do what is meant to be done.

2.10.1 Raspberry Pi Zero

- ☐ 1GHz single-core ARMv6 CPU (BCM2835)
- ☐ VideoCore IV GPU, 512MB RAM
- ☐ Mini HDMI and USB on-the-go ports
- ☐ Micro USB power
- ☐ HAT-compatible 40-pin header
- ☐ Composite video and reset headers
- ☐ CSI camera connector
- ☐ 802.11n wireless LAN
- ☐ Bluetooth 4.0

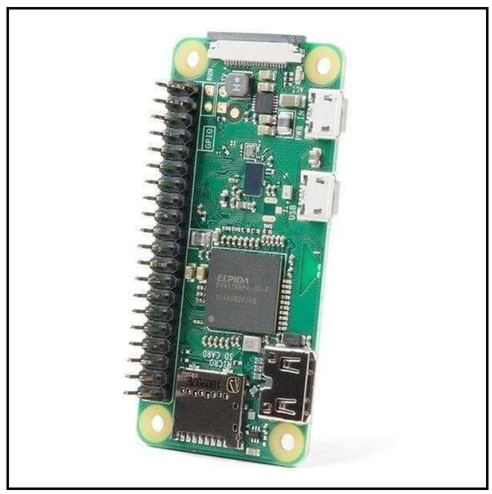


Figure 2.36 :Raspberry pi 0

2.10.2 STPM33 EVAL

- ☐ 0.2% accuracy single-phase meter with tamper monitoring
- ☐ $V_{nom(RMS)}$= 140 to 300 V, $I_{nom}/I_{max(RMS)}$= 5/100 A, f_{lin}= 50/60 Hz ± 10%
- ☐ Tamper detection through neutral current monitoring
- ☐ Connector for USB isolated hardware programmer tool
- ☐ SPI/UART switch for device peripheral selection
- ☐ 2 programmable LEDs on board
- ☐ Digital expansion to external system-on-chip or MCU
- ☐ 3.3 V power supply: external or through STEVAL isolated

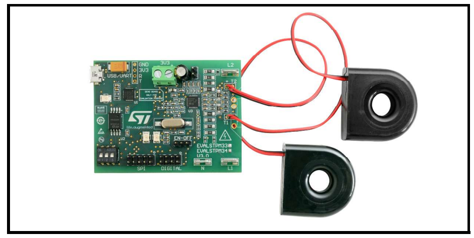


Figure 2.37: STPM33 EVAL

2.10.3 Hi-link Power Supply

- ☐ Ultra-thin, ultra-small, the industry's smallest volume
- ☐ Global universal input voltage (90 ~ 245Vac)
- ☐ Good output short circuit and overcurrent protection and self recovery
- ☐ Input and output isolation voltage
- ☐ Economic solutions, cost-effective
- ☐ No external circuit to work
- ☐ Low ripple, low noise
- ☐ High reliability, long life design, continuous working time is greater than 100,000 hours

Figure 2.38: HiLink Power Supply

2.10.4 Limit Switch

Figure 2.39:Limit Switch

☐ **3 pin Limit Switch Lever Arm**

☐ **Actuator**: This is the part of the switch that contacts the target.

☐ **Switch Body** : This part contains the electrical contact mechanism.

☐ **Pins:** Common , Normally Closed (NC) , Normally Open

2.10.5 TFT Display

Figure 2.40:TFT Display

☐ Display Size: 1.8"

☐ Driver IC: ST7735

☐ Input Voltage (V): 3.3 to 5

☐ Pixel Resolution: 128 x 160

☐ can display full 18-bit color

☐ PCB Size (mm): 62×38 (LxW)

☐ Interface Type: SPI

2.10.6 Push Button

Figure 2.41: Push Button

☐ Through-hole design.

☐ Shaft Shape: Round

☐ The long life of 300,000 cycles realized despite the high operating force

☐ Used in the fields of electronic products, household appliances and more.

☐ This light touch switch is waterproof, prevents oil, anti-pollution, and anti-static interference.

☐ High precision mechanism design offers acute operation and long service life.

☐ Compact and lightweight, easy to carry and dismantle.

☐ Good electrical conductivity.**[16]**

2.10.7 Mangopi F1c200s EVAL

☐ SoC – Allwinner F1C200s ARM926EJS processor @ 420 MHz (overclockable to 700 MHz) with 64MB DDR RAM

☐ Storage – 128MB NAND flash and MicroSD card slot

☐ Display I/F – 40-pin RGB565 display interface and 6-wire touch interface

☐ Camera I/F – 24-pin DVP camera interface compatible with OV2640, GC0328, etc.

☐ Audio – Onboard microphone; Class D audio amplifier

☐ Connectivity – Optional WiFi via TF-WiFi card (MicroSD WiFi card)

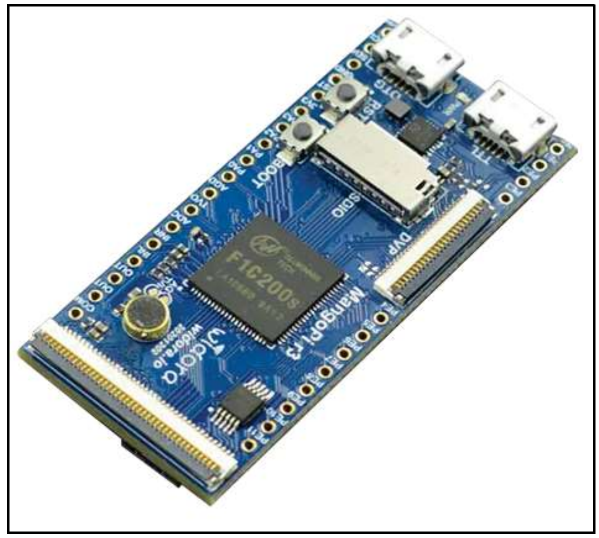


Figure 2.42: F1c200s EVAL

Micro USB port connected to CH340E USB-TTL serial controller

☐ Expansion – 30 through holes on the sides for GPIOs, I2C, SPI, SDIO, audio output/input, etc...

☐ Misc – Boot & Reset buttons

☐ Power Supply – 5V via Micro USB port

☐ Dimensions – 50.8 x 22.86mm

☐ USB – 1x Micro USB OTG port also supporting FEL flash mode

☐ Debugging –

2.11 Edge Computing and DCU-Less Concept

Introduction

In the present age of innovation-driven world, we have IoT devices, 5G fast wireless, mobile applications and smart dashboards, all are generating enormous amounts of data and sent out into the Cloud for processing these mass information into helpful information.

Issues associated

But there are some issues & problems with this setup, as these sensors don't transmit data regularly and security concerns are associated as well when data acquisition and processing are separated by hundreds of miles.
Likewise, what happens if the Internet link of the facility gets interrupted somehow? Cloud would certainly be worthless then.
Here comes EC (Edge Computing) to rescue you. Let's discuss this

What is Edge Computing?

Simple version, it is a computer that resides close to your sensors, collects data, and can do data processing. This way, instead of transporting all that data to the Cloud or to your data center, you deal with it locally first.

The "computing" part can be as tiny as a Raspberry Pi to a rack full of servers with multiple GPUs running the latest neural networks. **[17]**

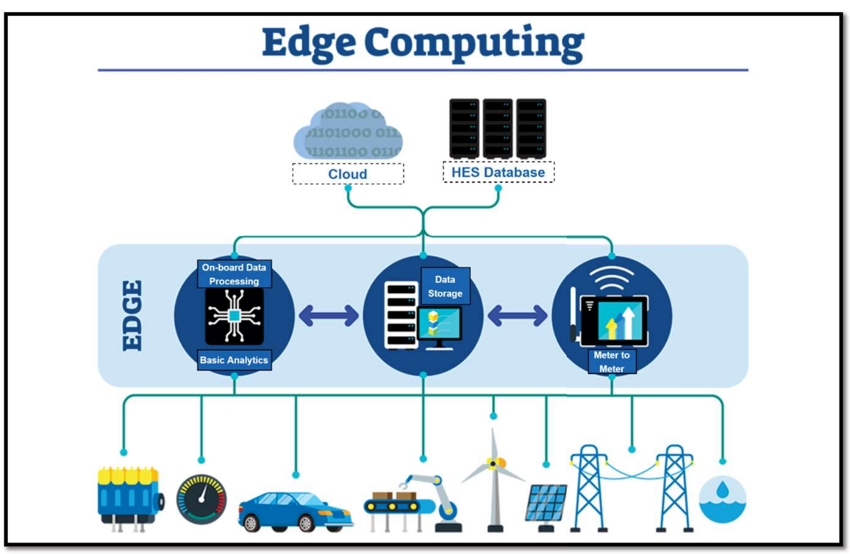


Figure 2.43: Edge Computing Principle around us

Working Mmechanism

By this, you can do refining as close to data acquisition as possible. This not only saves cost of data transfer expenses and also saves storage cost, if you can clean up the data before uploading it, since you do not require 100% of the video clip.

Edge Computing allows you to process sensor data before involving a data center.

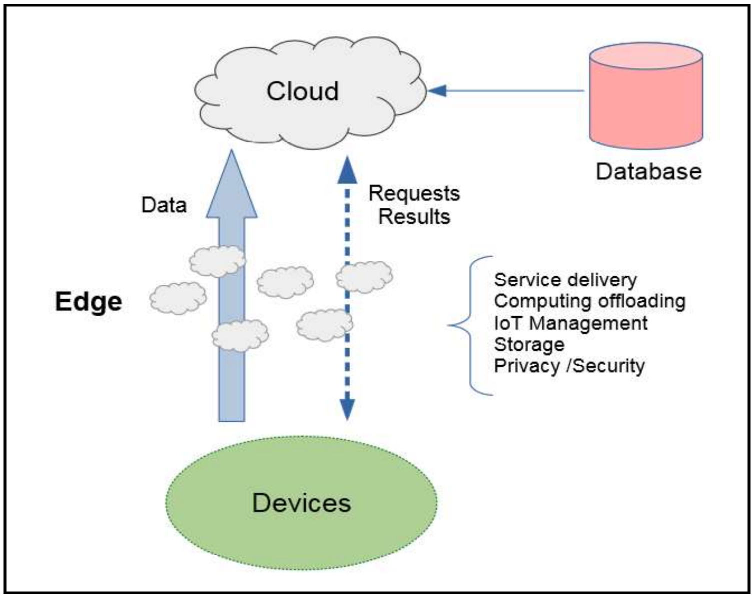


Figure 2.44: Edge Computing

Another benefit to think about: if your network goes down, the edge device(s) can store the data until network connectivity is restored and upload it at that point.

One way to solve this is to use an edge device. The sensors submit their data to this device and the device manages to upload the data to your data center. If there is no Internet connectivity, you can generate a report from the readings stored on the edge device.

Edge Hardware

The selection of hardware is dependent on the task you are doing either store-and-forward of the data or do you need complex computing needs.
The difference can be a $30 Raspberry Pi or a rack full of GPUs that's > $10,000 a piece.

Edge systems are just another production system

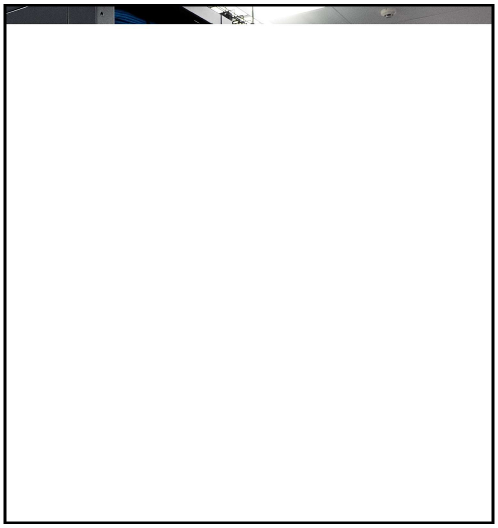


Figure 2.45: Data Centers

Application of Edge Computing

Edge application services minimize the volumes of data that have to be relocated, as well as the range that data have to take a trip. That gives lower latency as well as lowers transmission expenses. Computation offloading for real-time applications, such as facial recognition algorithms.

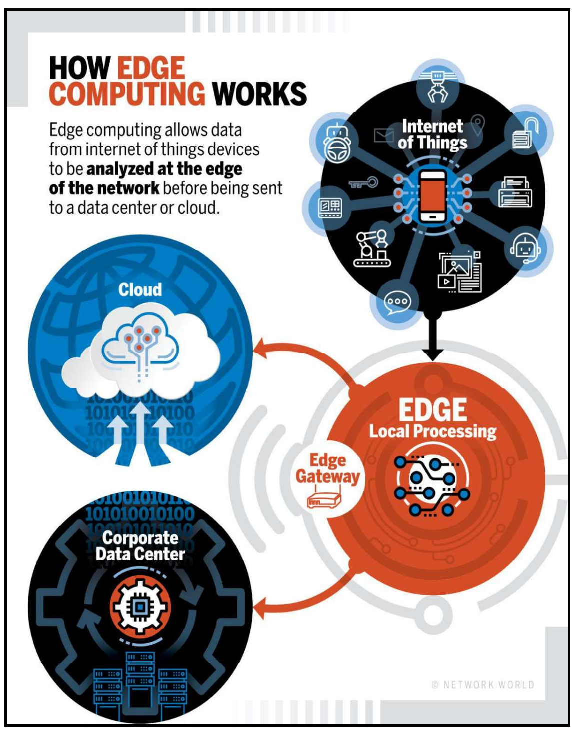


Figure 2.46: Network Cloud

Cloud gaming is another notable use-case, where some aspects of a game could run in the cloud, while the rendered video is transferred to lightweight clients running on devices such as mobile phones, VR glasses, etc. This type of streaming is also known as pixel streaming.

Other famous applications include connected cars, autonomous cars, smart cities, Industry 4.0 (smart industry), and home automation systems.

Advantages of Edge computing

Edge computing has several advantages, such as:

- Increasing data security and privacy
- Better, more responsive and robust application performance
- Reducing operational costs
- Improving business efficiency and reliability
- Unlimited scalability
- Conserving network and computing resources
- Reducing latency

Challenges in deploying Edge computing
Some major challenges people face while applying Edge computing include:

- Remote connectivity and debugging, where multiple devices need to be identified and connected.
- Model, firmware, and data up-gradation, as video analytics require machine learning model updates and some gateways need firmware upgrades.
- Lack of trained personnel to manage the devices due to complexity of systems and technical barriers.

Ending Lines
Edge computing is essential and widely used especially with the advent of 5G and the rise of IoT. These include (but are not limited to) connectivity, bandwidth management, sparse/bad data, heavy computing needs, etc.

Traditional Electricity Metering

The electrical devices that can detect and display energy in the form of readings are termed as electricity meters. Traditional meters have been used since the late 19th century. They exchange data between electronic devices in a computerized environment for both electricity production and distribution. In most of the traditional electricity meter aluminum discs are used to find the usage of power. Today's electricity meter is digitally operated but still has some limitations.

Some of the limitations faced by the traditional electricity meter are as follows:

- Meters are unreliable in nature as consumers have to anticipate the monthly electricity bill.
- The process of measurement is supported by a specific mechanical structure and hence they are called electromechanical meters.
- In order to perform meter readings, a great number of inspectors have to be employed.
- Payment processing is expensive and time consuming.

Embedded Linux

In order to solve these problems, We prefer using embedded linux rather than embedded systems because any meter will contain a specific DCU. A Data Concentrator Unit (DCU) is deployed at the data perception layer of the AMI architecture. It can function as the IoT gateway to implement communication between smart meters and the application system, and can also serve as a data collection device to execute instructions to collect data of smart meters. It implements data exchange between terminals and the application system. Moreover, Linux is also used heavily in many production environments. Embedded Linux is the use of Linux in embedded computer systems such as mobile phones, personal digital assistants, media players, set-top boxes and other consumer electronics devices, networking equipment, machine control, industrial automation, navigation equipment and medical instruments which have:

- Low Memory Resources – RAM, Hard Disk
- Low Processing Power – CPU (400 MHz, <= 1GHz)
- Compact Size
- Low Cost Limitation
- Higher Performance and Efficiency in Speed, Power

2.12 Azure Cloud

Microsoft Azure, formerly known as Windows Azure, is Microsoft's public cloud computing platform. It provides a range of cloud services, including compute, analytics, storage and networking. Users can pick and choose from these services to develop and scale new applications, or run existing applications. We use Azure cloud to deploy our on-cloud services like the APIs and the Web application.

Figure 2.47: Microsoft azure

The Azure platform aims to help businesses manage challenges and meet their organizational goals. It offers tools that support all industries including e-commerce, finance and a variety of Fortune 500 companies and is compatible with open source technologies. This provides users with the flexibility to use their preferred tools and technologies. In addition, Azure offers four different forms of cloud computing:

- infrastructure as a service (IaaS).
- platform as a service (PaaS).
- software as a service (SaaS).
- Serverless.

How does Microsoft azure work?

Once customers subscribe to Azure, they have access to all the services included in the Azure portal. Subscribers can use these services to create cloud-based resources, such as virtual machines (VM) and databases. In addition to the services that Microsoft offers through the Azure portal, a number of third-party vendors also make software directly available through Azure. The cost billed for third-party applications varies widely but may involve paying a subscription fee for the application, plus a usage fee for the infrastructure used to host the application.

What is Microsoft Azure used for?

Because Microsoft Azure consists of numerous service offerings, its use cases are extremely diverse. Running virtual machines or containers in the cloud is one of the most popular uses for Microsoft Azure. These compute resources can host infrastructure components, such as domain name system (DNS) servers; Windows Server services -- such as Internet Information Services (IIS); or third-party applications. Microsoft also supports the use of third-party operating systems, such as Linux.

2.13 Database

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just a database. Data within the most common types of databases in operation today is typically modeled in rows and columns in a series of tables to make processing and data querying efficient. The data can then be easily accessed, managed, modified, updated, controlled, and organized. Most databases use structured query language (SQL) for writing and querying data.

What's the difference between a database and a spreadsheet?

Databases and spreadsheets (such as Microsoft Excel) are both convenient ways to store information. The primary differences between the two are:

- How the data is stored and manipulated
- Who can access the data
- How much data can be stored

Spreadsheets were originally designed for one user, and their characteristics reflect that. They're great for a single user or small number of users who don't need to do a lot of incredibly complicated data manipulation. Databases, on the other hand, are designed to hold much larger collections of organized information in massive amounts, sometimes. Databases allow multiple users at the same time to quickly and securely access and query the data using highly complex logic and language.

Types of databases:

There are many different types of databases. The best database for a specific organization depends on how the organization intends to use the data.

- **Relational databases:** Relational databases became dominant in the 1980s. Items in a relational database are organized as a set of tables with columns and rows. Relational database technology provides the most efficient and flexible way to access structured information.
- **Object-oriented databases:** Information in an object-oriented database is represented in the form of objects, as in object-oriented programming.
- **Distributed databases:** A distributed database consists of two or more files located in different sites. The database may be stored on multiple computers, located in the same physical location, or scattered over different networks.
- **Data warehouses:** A central repository for data, a data warehouse is a type of database specifically designed for fast query and analysis.
- **NoSQL databases:** A NoSQL, or non relational database, allows unstructured and semistructured data to be stored and manipulated (in contrast to a relational database, which defines how all data inserted into the database must be composed). NoSQL databases grew popular as web applications became more common and more complex.
- **Graph databases:** A graph database stores data in terms of entities and the relationships between entities.
- **OLTP databases:** An OLTP database is a speedy, analytic database designed for large numbers of transactions performed by multiple users.

2.14 GUI Applications

The term "GUI" stands for "graphical user interface," which refers to the user interface that enables people to interact visually with electronic devices like computers, laptops, smartphones, and tablets. In terms of human-computer interaction, it's an important component of software application programming to swap out text-based commands with user-friendly actions. The user will be given choice points that are simple to identify, comprehend, and employ. In other words, GUI enables you to use a mouse, pen, or even your finger to control your gadget.
Because text command-line interfaces were complex and challenging to understand, GUI was developed. A command or function on your devices, such as tabs, buttons, scroll bars, menus, icons, pointers, and windows, can be

opened by clicking or pointing to a small image, known as an icon or widget, through the GUI process. Today, user-centered design for software application programming is the norm.

GUI elements:

1. <u>Input Control:</u>

Buttons: Buttons are circles that let you make immediate choices and take actions. Radio buttons come in groups where only one button can be selected at a time. Label buttons have text on them. If you want more than one option to be selected, consider using a check box.

Checkboxes: Checkboxes are square boxes in a list of one or more options. When you click the box, it stays selected. They are best presented in a vertical list. A checkbox can be a single box, such as acknowledging a statement, or a list of related items, such as a shopping list.

Date picker: A date picker lets you select a date and/or time. The creator can choose a calendar or a fill-in option. It ensures that a consistent format is used, such as "day, month, year."

Dropdown lists: Dropdown lists lets you select one item at a time. Several items can be included compactly. Consider adding directions, such as "select one" to let the user know what to do. The creator can add or delete items to keep the list up to date.

List boxes: List boxes let you select multiple items from one compact list. Use this GUI feature if you have a long list of options for the user to consider. There are four variations of list boxes: single-line, multiselect, multiselect with checkboxes and multiselect-dual list boxes.

Text boxes: Text boxes are fields that let you enter text. The creator can control how much text is allowed.

2. <u>Navigational components:</u>

Icons: An icon is a small image used as a symbol to help you navigate the system. They are typically used to indicate an application, folder, file or web browser. Using an icon is a fast way to open documents and run programs. Also, all files that you create in the same application will have the icon of the application and the same extension.

Tabs: A tab is a small box that displays the name or graphical icon associated with a specific window. When you choose a tab, you will see the specific controls and information presented in that window. For example, when you open multiple pages in a web browser, you will see the different tabs displayed at the top of the browser window.

3. Informational components:

Message box: A message box is a small window with information such as a policy or disclaimer. It requires you to take action before you proceed.

Notifications: A notification is a message box. Typically, they are used to indicate emergency warnings, error messages or task completion.

Benefits of GUI:

GUI uses visual elements to represent those now hidden lines of command. You simply select a button or an icon to call the relevant function. The easy use of GUIs has made it possible for the public in general, regardless of experience or knowledge, to access all kinds of systems for everyday use.

- **Easy to use:**

Since data is represented by symbols, shapes and icons, users can easily recognize, classify and navigate options. A simple click is all it takes to acquire a function. Because it's so easy to use and understand, GUI has become the preferred interface for computers and mobile devices.

- **Easy to communicate:**

Visual representation of data is recognized faster than text. Non-programmers find it easy to use GUIs since it requires no experience with computing commands. They don't have to worry about writing or debugging code. As a result, users find GUI an easy-to-learn interface.

- **Attractive:**

GUI has visually appealing features and is not cluttered with command line codes. Visual images can portray emotions, comments and situations with long lines of computer language. Pictures and such are easy to understand and often carry universal meaning.

- **Allow for multitasking**

GUI lets users work and view two or more programs at the same time. For example, you can view a streaming presentation while searching the internet from a web browser. You can watch a video while writing a review of the presentation with a search engine in another tab.

2.15 Machine Learning

- **Overview**

Machine learning is a data analysis technique in which computers are taught to make decisions based on experience. With the increase in the amount of big data, machine learning has become a crucial technique to solve problems.

Classical machine learning is often categorized by how an algorithm learns to become more accurate in its predictions. There are two basic approaches:

1. **Supervised machine learning algorithms:**

It applies what has been learned in the past to new data using labeled examples to predict future events. By analyzing a known training dataset, the learning algorithm produces an inferred function to predict output values. The system can provide targets for any new input after sufficient training. It can also compare its output with the correct, intended output to find errors and modify the model accordingly. Common problems include:

i. Regression: This is a method for understanding the relationship between independent variables or features and a dependent variable or outcome.
ii. Classification: which is defined as the process of recognition, understanding, and grouping of objects and ideas into preset categories a.k.a “sub-populations.”

2. Unsupervised machine learning algorithms:

They are used when the information used to train is neither classified nor labeled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. At no point does the system know the correct output with certainty. Instead, it draws inferences from datasets as to what the output should be. An example on that would be Clustering algorithms.

3. Reinforcement Learning

It is used to teach a machine to complete a multi-step process for which there are clearly defined rules. Data scientists program an algorithm to complete a task and give it positive or negative cues as it works out how to complete a task. But for the most part, the algorithm decides on its own what steps to take along the way. **[18]**

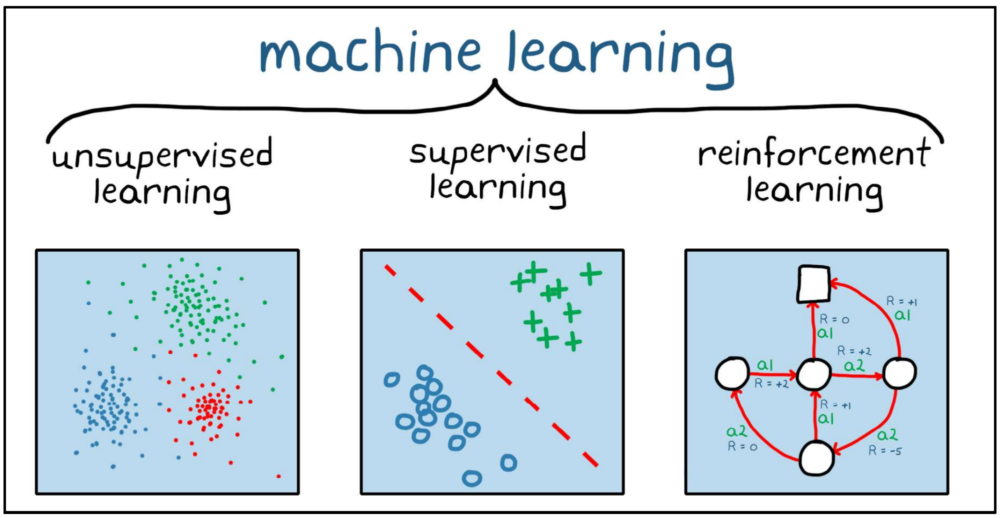


Figure 2.48: Classification of machine learning algorithms

- Deep Learning

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. The core of deep learning is that we now have fast enough computers and enough data to actually train large neural networks. **[19]**

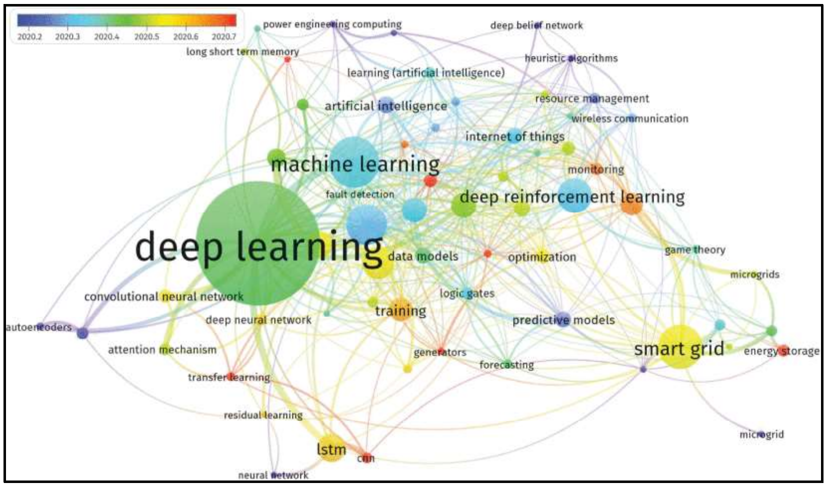


Figure 2.49: Deep Learning

- **ANN**

Artificial neural networks (ANNs) are composed of a node layer, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network. Neural networks rely on training data to learn and improve their accuracy over time.

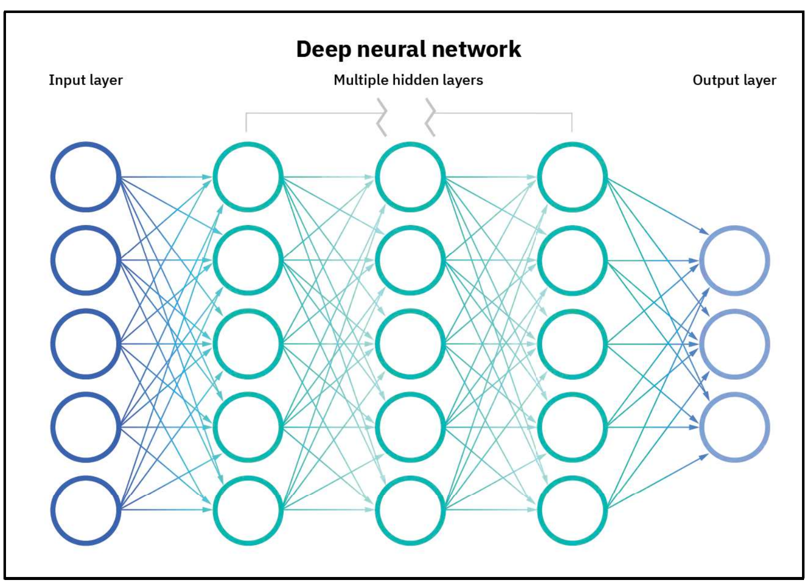


Figure 2.50: Typical structure of neural network

- **CNN**

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in a 2D matrix (or 1D Vector), assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The model learns to extract features from sequences of observations and how to map the internal features to different activity types. The benefit of using CNNs for sequence classification is that they can learn from the raw time series data directly, and in turn do not require domain expertise to manually engineer input features.

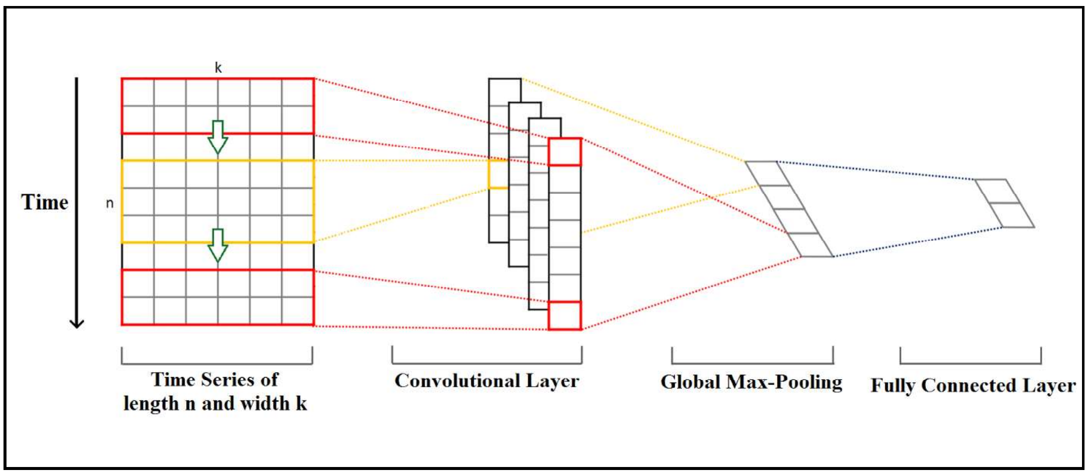


Figure 2.51: 1D CNN

- **RNN**

A recurrent neural network (RNN) is a type of artificial neural network which uses sequential data or time series data. These deep learning algorithms are commonly used for ordinal or temporal problems, such as language translation, natural language processing (nlp), speech recognition, and image captioning. Like feedforward and convolutional neural networks (CNNs), recurrent neural networks utilize training data to learn. They are distinguished by their “memory” as they take information from prior inputs to influence the current input and output. While traditional deep neural networks assume that inputs and outputs are independent of each other, the output of recurrent neural networks depend on the prior elements within the sequence. While future events would also be helpful in determining the output of a given sequence.

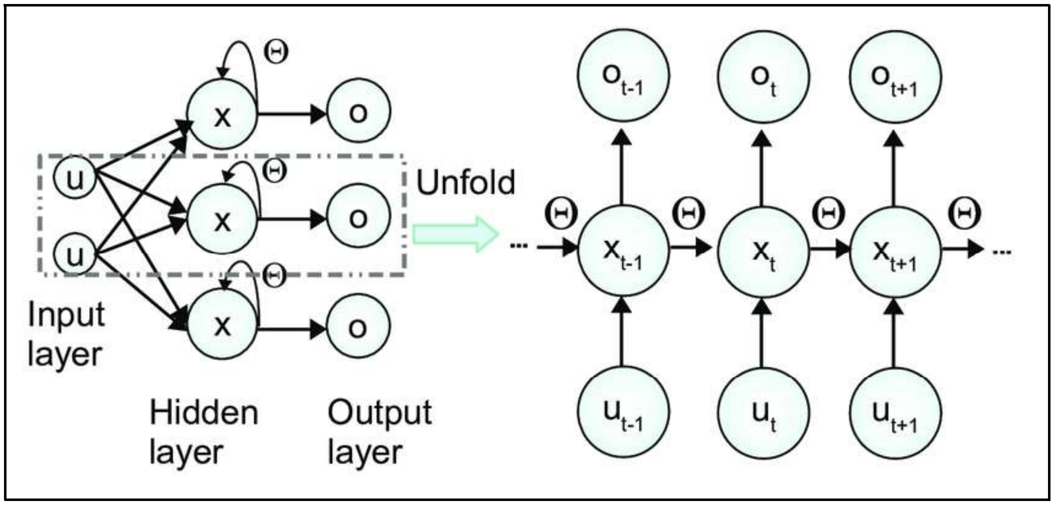


Figure 2.52: RNN

- **ML in Smart Grid**

Smart grid is a cyber-physical-social system where the power flow, data flow, and business flow are deeply coupled. Enlightened consumers facilitated by smart meters form the foundation of a smart grid. Countries around the world are in the midst of massive smart meter installations for consumers on the pathway toward grid digitalization and modernization. It enables the collection of extensive fine-grained smart meter data, which could be processed by data analytical techniques, especially now widely available machine learning techniques.

Machine learning techniques provide an efficient way to analyze, and then make appropriate decisions to run the grid; and thus enable the smart grid to function as it is intended to. Machine learning functionalities include:

- Predictions of consumption price
- Power generation
- Future optimum scheduling
- Fault detection
- Detection of network intruders and
- Load disaggregation

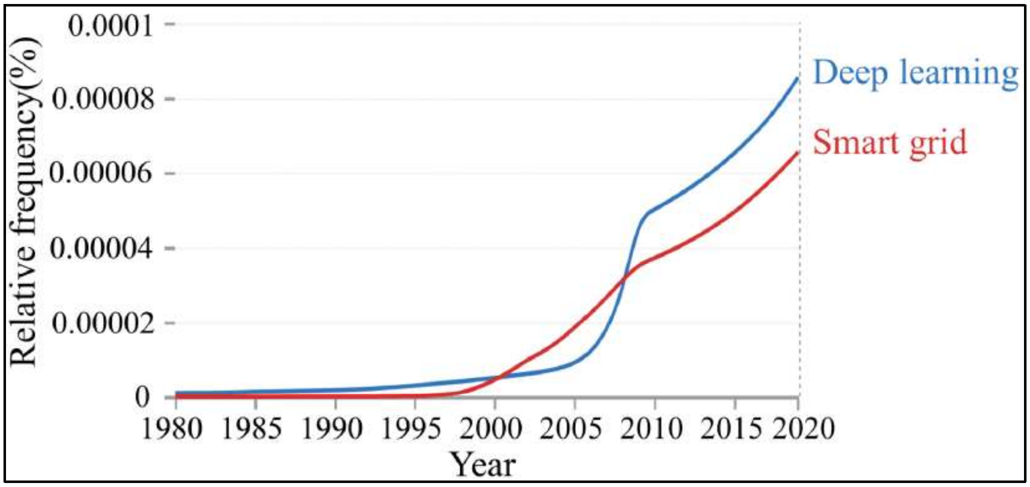


Figure 2.53: Frequency of use of terms deep learning and smart grid in books from 1980 to 2020

3. NGSM System Architecture

3.1 System Architecture Overview

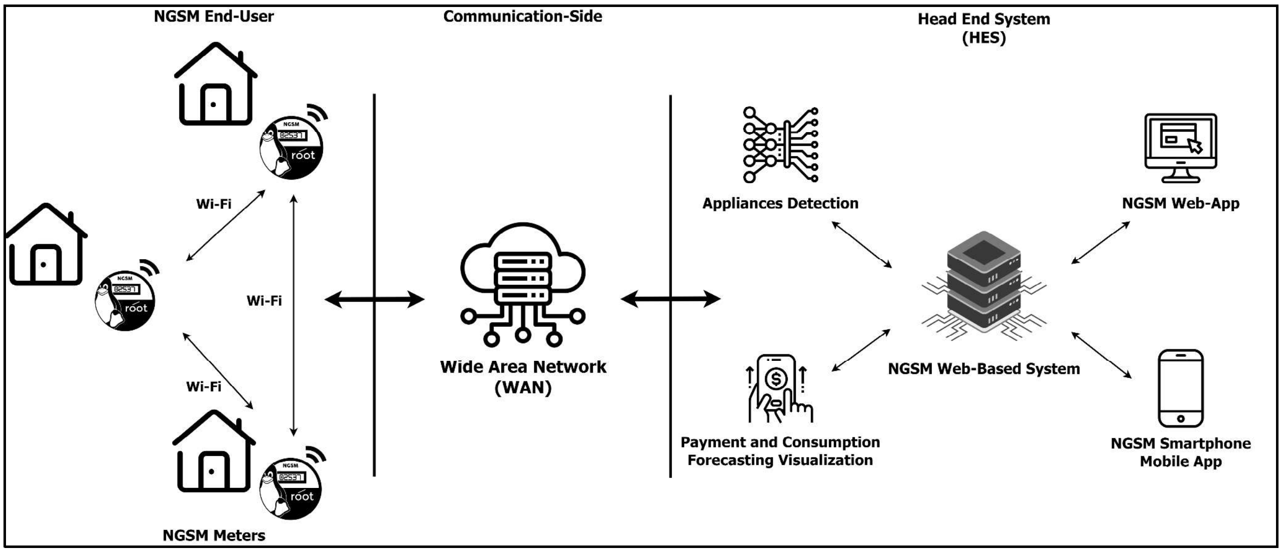


Figure 3.1: NGSM System Block Diagram

-Idea is to deliver complete Advanced Meter Infrastructure (AMI) that include:

Smart Meter

Embedded Linux device which is Combination of Software and Hardware contains on System on Chip (SoC) capable of running Linux with low power consumption and appropriate Clock Speed, permits to Process on the Meter itself (Edge Computing) and extract the required features for sending it to the Head End System instead of the real time Communication for sending the raw data to process on the Server Side. The Meter enables 2 ways of communication: Power Line Communication Modem which is very affordable and cheap and Wireless Communication every N of Meters will communicate and share their data in the MESH Network by PLC and their Master will send the Data to the Head End System. **[20]**

Head End System

- <u>Mobile Application</u>

It's used for the interface between the End user (Electricity consumer) and the Head End System ,and it's also used for payment of the bill, Monitor the Profile and Dashboard of the usage.

- Web System

Web System Enable to Communicate with the user (Electricity consumer), and show the load profiles, Control the Metering System, pay the bill

3.2 System Components Features

1- NGSM Electricity Meter:

- Measure and calculate all the required electric usage parameters, from volt and current to energy and power.
- Send the collected data to the HES for further usage processing and calculating the bill.
- Detecting tampering.
- Provides customers with greater control over their electricity use when coupled with time-based rates.
- Allows customers to make informed decisions by providing detailed information about the working household appliances.

2- NGSM Web-based System and Smartphone Mobile Application:

- Receives and stores meter measurements and events.
- Calculates the user's bill.
- Visualized hourly consumption.
- 100% online and remote payment.
- Updates users with the working household appliances and consumption forecasting.

3.3 Smart Meter Hardware Architecture

Hardware Architecture is divided into main five blocks; 1st analog Front End (AFE), 2nd Single Board Computer (SBC), 3rd User Interaction, 4th Enclosure Tampering Detection and 5th Power Supply, each block of them contains sub-blocks are demonstrated as follows:

1. AFE contains the main sensing elements for measuring voltage and current, Power Measurements unit & Digital Signal Processing unit (DSP) and the Electricity Tampering Detection which is regularly done by measuring the live and neutral current (not only the live current) to check the equality of them, if they aren't equal so tampering will be detected.

2. SBC contains the main Controller which is the Powerful System on Chip (SoC) has the ability to run Linux beside storage such as SD card and Modem to provide connectivity over the network.
3. User Interaction contains the graphical display which shows the consumption for the user, buttons to control the display and moving between its pages and LED for indication.
4. Enclosure Tampering Detection which regularly uses Limit Switch is pressed by the cover, if someone tries to disassemble the cover, that will cause releasing the limit switch and tampering will be detected.
5. Power Supply which is used to supply all components with the required Power.

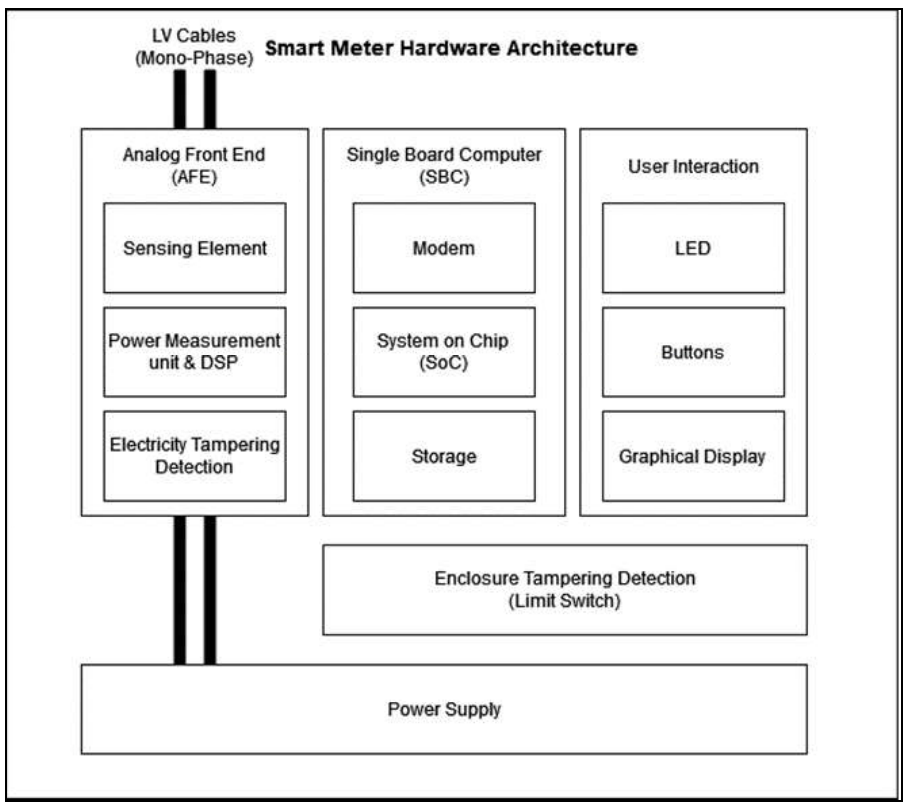


Figure 3.2 : Smart Meter Hardware Architecture

3.4 Smart Meter Software Architecture

Software Architecture in Embedded Linux differs from the BareMetal Embedded systems as you're dealing with Linux to control your hardware. A modern computer OS usually divides virtual memory into user space and kernel space. Primarily, this separation serves to provide memory protection and hardware protection from malicious software.

User space refers to all code that runs outside the operating system's kernel. It contains the various user programs and libraries that the operating system uses to interact with the kernel: software that performs input/output, manipulates file system objects, application software, etc.

Kernel space is strictly reserved for running the kernel (the core of the operating system) and providing its services and most device drivers which control the hardware. **[21]**

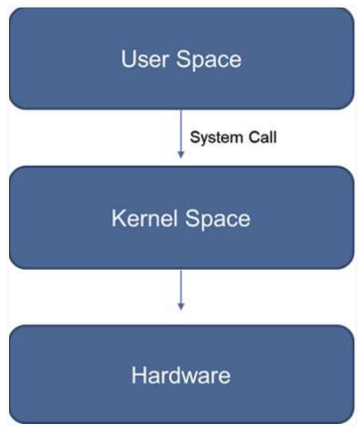


Figure 3.3: Design in Embedded Linux

When you're dealing with Linux you can write your program in User-Space or Kernel-Space as a kernel module which may be faster due to less context switching, less system call overhead, and less interruptions, and certainly do run at very high priority. On the other side there're drawbacks such as; Debugging gets harder, errors become more frequent and harder to find. You might compromise security and stability. So, it's recommended to let your code in user-space, unless it's **really** necessary to write in kernel space. That's why we wrote all programs in the user space which provides more security and satisfies our needs.

The Software architecture is divided into sub-Programs run in the user space as daemons at the startup of Linux.

- **System Processes:**

- **Events daemon**

Meter Events contains Tampering Detection which occurs when there is unauthorized removal of a meter, severing of a meter seal, opening of a meter base, altering an entrance cable in any manner, or self-reconnects that are not done by an authorized PSE employee or representative.

- **Measurements daemon**

Measures and records information such as consumption of electric energy, voltage levels, current, and power factor. Smart meters communicate the information to the consumer for greater clarity of consumption behavior, and electricity suppliers for system monitoring and customer billing.

- **Display daemon**

Sending commands over SPI protocol for the Display controller to display the specific screens based on the user button's state, also receives the incoming from the Measurements daemon.

- **Server and modem daemon**

By implementing Web Server on the Meter will provide the ability to send and receive requests from the HES also it can apply HTTP requests like Get & Post. to achieve that, Modem is used to access the internet. We used for the Web Server gSOAP Library which provides the ability to update the HES with data. gSOAP is written in C/C++, that’s why it’s lite and fast which makes it a better option to be used in Embedded devices.

- **Database**

Simple database which contains the Readings & Events with Timestamp. It’s used to update the data to the cloud by scheduling.

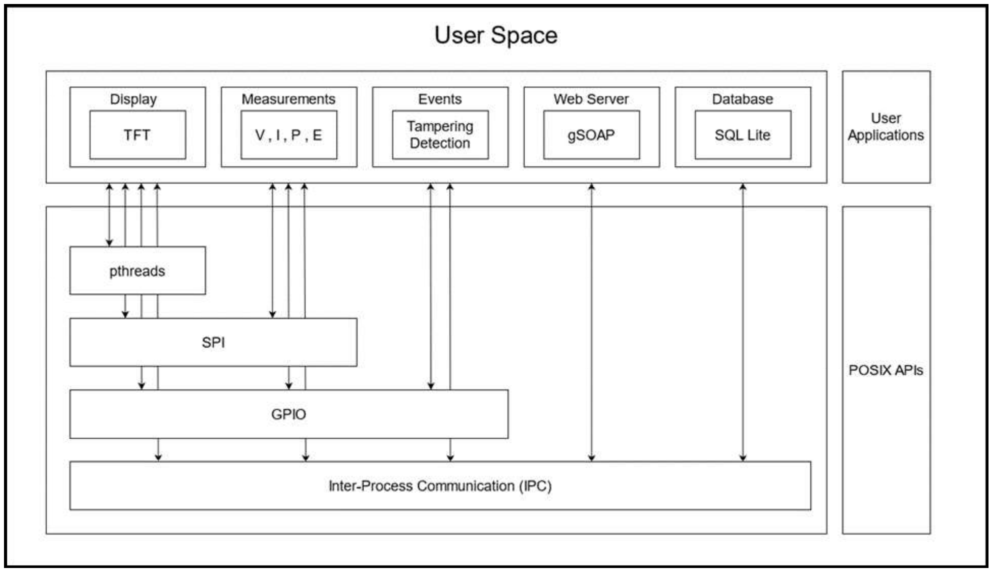


Figure 3.4: User-Space applications in Linux

4. 1st Generation NGSM Smart Meter

4.1 Hardware Specification

The 1st generation uses the main SoC of Raspberry-Pi 3 and it has the sufficient processing power for running our Software Implementation. We used this selection to fit our implementation. It has a 64-bit quad-core ARM Cortex-A53 which is the Broadcom BCM2837 SoC clocked at 1.2GHz, supporting armv7 architecture with 1 GB LPDDR2 and also supporting Wi-Fi & Bluetooth.

We used STPM33 as an AFE unit, it supports UART and SPI, we selected SPI as it's faster to communicate with STPM33.

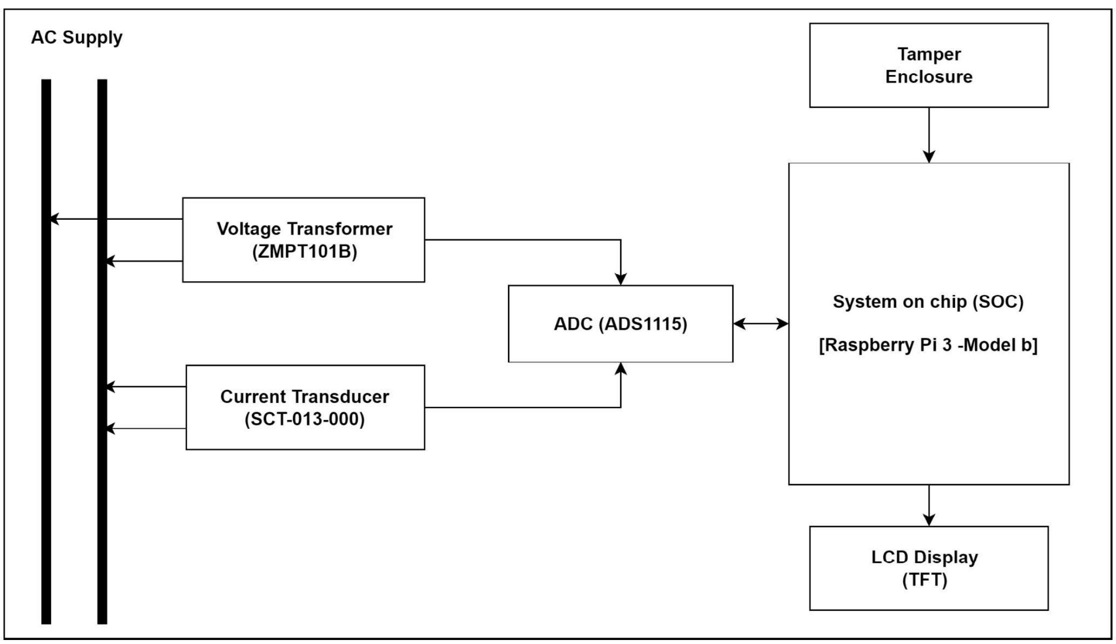


Figure 4.1: 1st Generation NGSM Hardware Components

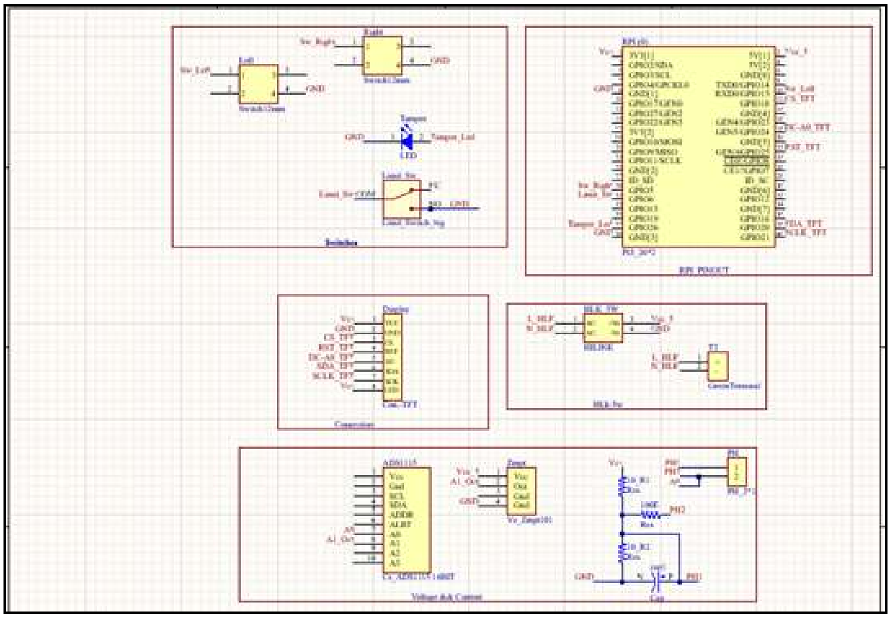


Figure 4.2: PCB Schematic

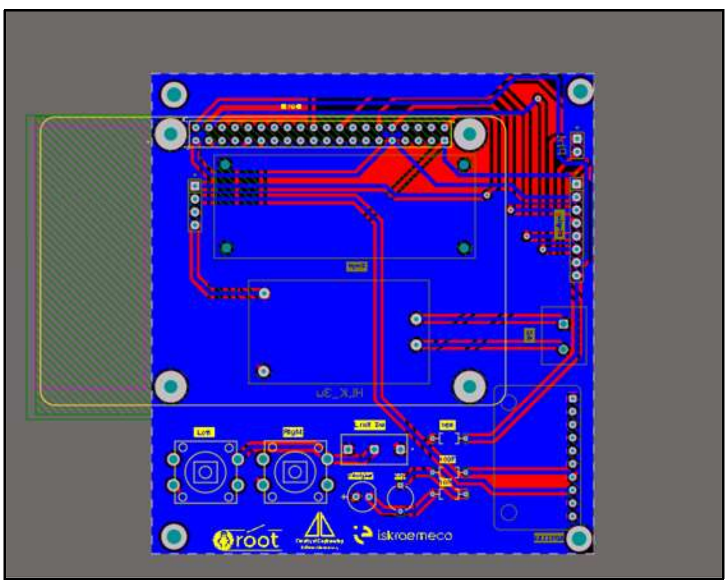


Figure 4.3: PCB Layout

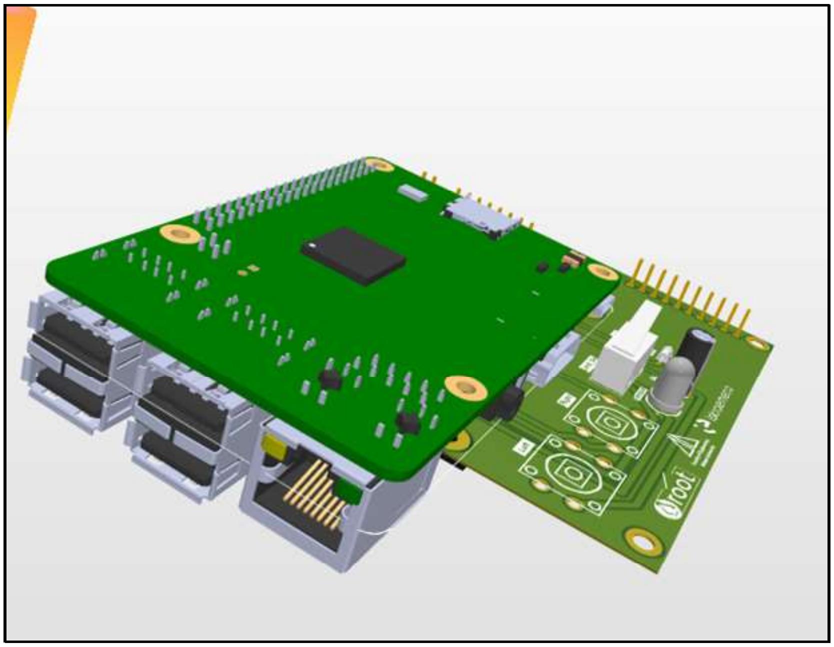


Figure 4.4: PCB 3D Model

4.2 Software Implementation

Software Architecture in Embedded Linux differs from the BareMetal Embedded systems as you're dealing with Linux to control your hardware. A modern computer OS usually divides virtual memory into user space and kernel space. Primarily, this separation serves to provide memory protection and hardware protection from malicious software.

User space refers to all code that runs outside the operating system's kernel. It contains the various user programs and libraries that the operating system uses to interact with the kernel: software that performs input/output, manipulates file system objects, application software, etc.

Kernel space is strictly reserved for running the kernel (the core of the operating system) and providing its services and most device drivers which control the hardware. **[22]**

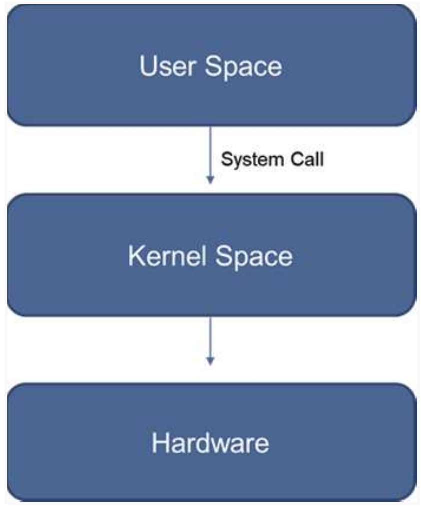


Figure 4.5: Design in Embedded Linux

When you're dealing with Linux you can write your program in User-Space or Kernel-Space as a kernel module which may be faster due to less context switching, less system call overhead, and less interruptions, and certainly do run at very high priority. On the other side there're drawbacks such as; Debugging gets harder, errors become more frequent and harder to find. You might compromise security and stability. So, it's recommended to let your code in user-space, unless it's really necessary to write in kernel space. That's why we wrote all programs in the user space which provides more security and satisfies our needs.

The Software Architecture is divided into sub-Programs run in the user space as daemons at the startup of Linux.

System Processes

- Events daemon

Meter Events contain Tampering Detection which occurs when there is unauthorized removal of a meter, severing of a meter seal, opening of a meter base, altering an entrance cable in any manner, or self-reconnects that are not done by an authorized PSE employee or representative.

- Measurements daemon

Measures and records information such as consumption of electric energy, voltage levels, current, and power factor. Smart meters communicate the information to the consumer for greater clarity of consumption behavior, and electricity suppliers for system monitoring and customer billing.

- Display daemon

Sending commands over SPI protocol for the Display controller to display the specific screens based on the user button's state, also receives the incoming message from the Measurements daemon.

- Server and modem daemon

Implementing a Web Server on the Meter will provide the ability to send and receive requests from the HES and can also apply HTTP requests like Get & Post. to achieve that, Modem is used to access the internet. We used the Web Server gSOAP Library which provides the ability to update the HES with data. gSOAP is written in C/C++, that's why it's lite and fast which makes it a better option to be used in Embedded devices.

- Database

Simple database which contains the Readings & Events with Timestamp. It's used to update the data to the cloud by scheduling.

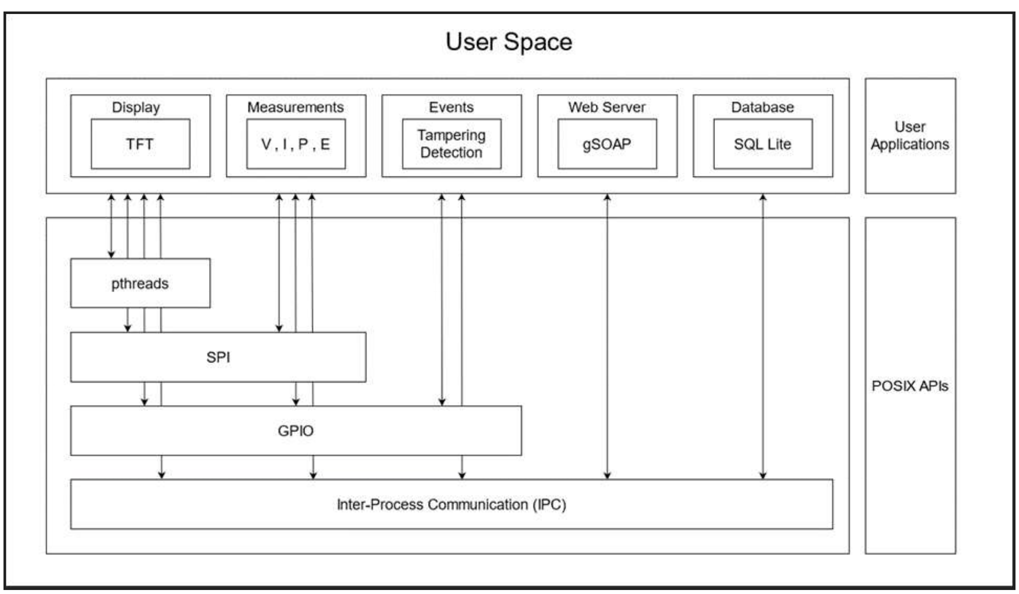


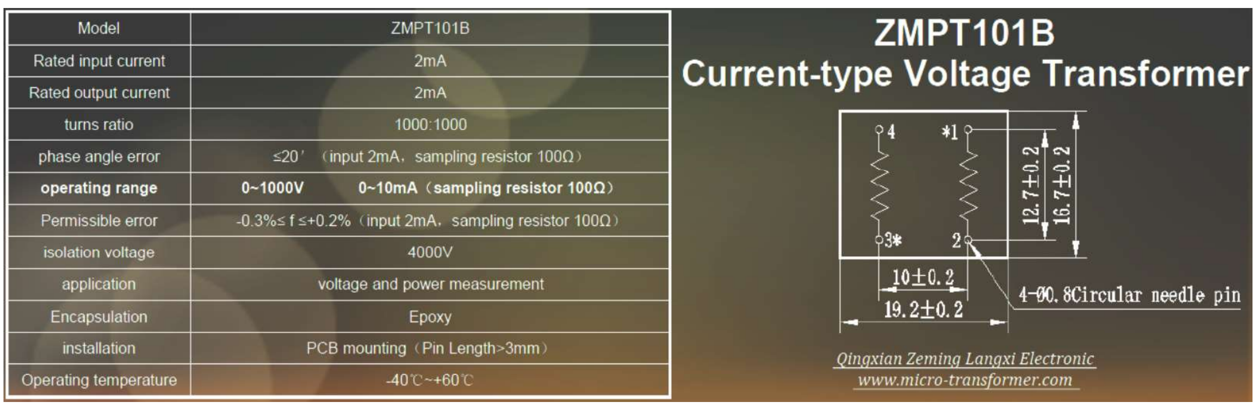
Figure 4.6: User-Space applications in Linux

4.3 Electricity Metering Procedure and Calculations

This generation uses two electricity sensors the ZMPT101B AC voltage sensor module and the SCT013 AC current sensor module:

ZMPT101B AC Voltage Sensor Module:

The ZMPT101B module is a compact single-phase AC voltage sensor module based on the minuscule 2mA/2mA precision voltage transformer ZMPT101B. Considering the ZMPT101B transformer alone, the setup is simple as it only needs a limit resistor on the primary side and a sampling resistor on the secondary side.



Model	ZMPT101B
Rated input current	2mA
Rated output current	2mA
turns ratio	1000:1000
phase angle error	≤20′ （input 2mA，sampling resistor 100Ω）
operating range	**0~1000V 0~10mA（sampling resistor 100Ω）**
Permissible error	-0.3%≤ f ≤+0.2%（input 2mA，sampling resistor 100Ω）
isolation voltage	4000V
application	voltage and power measurement
Encapsulation	Epoxy
installation	PCB mounting（Pin Length>3mm）
Operating temperature	-40℃~+60℃

Figure 4.7: Overview of the Transformer

The primary and secondary coils of this transformer have a DC resistance close to 110Ω at 20°C.

Coming back to the ZMPT101B module, it can handle AC voltages up to 250V (50Hz/60Hz). Its secondary circuitry, centered on the LM358 dual op-amp chip, also allows tweaking the isolated analog output via an onboard multi turn trimpot. The recommended operating voltage of the module is 5VDC.

ZMPT101B Module & Initial Output Signal Tweaks

In principle, the analog output signal of the module fluctuates up and down within predefined limits (when no ac input voltage is detected, it will deliver an output at half the supply voltage). Pay attention to ensure that the module's output is a complete sine wave. If the waveform is clipped, you must refine the output to head off potential measurement errors (naturally each module has its own deviations, so the onboard amplitude adjust trimpot will be a helpful inclusion). You need to keep an eye on the output waveform before first use.

SCT013 AC Current Sensor Module:

The SCT013 series are sensors of non-invasive, current transformers that measure the intensity of a current that crosses a conductor without needing to cut or modify the conductor itself. We can use these sensors with a processor, like Arduino, to measure the intensity or power consumed by a load.

The SCT013 sensors are current transformers, instrumentation devices that provide a measurement proportional to the intensity that a circuit crosses. The measurement is made by electromagnetic induction.

SCT013 sensors have a split core (like a clamp) that allows the user to turn it on to wrap electrical equipment without having to cut it off.

In the SCT013 series there are models that provide the measurement as a current or a voltage output. It is more preferable to use voltage output because the connection is simpler.

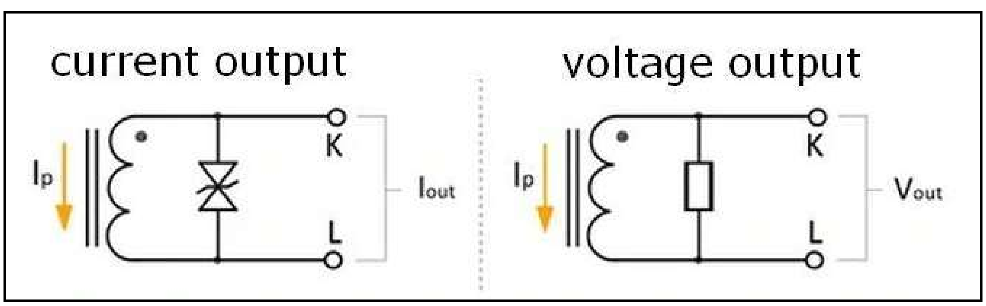


Figure 4.8: Current or a Voltage Output

The sensor accuracy can be off by only 1-2%. To ensure highest accuracy, it is critical to confirm that the core has been properly closed. Even a small air gap can cause a 10% deviation.

As a disadvantage, being an inductive load, the SCT013 introduces a variation of the phase angle, whose value is a function of the load that passes through it, being that it is able to reach up to 3°.

Current transformers are common components in the industrial world as well as in electrical distribution, as they allow the points of consumption to be monitored, whereas another form of measurement does not exist. They are also considered multiple measurement instruments, even in portable equipment such as perimeter clamps or network analyzers.

For example, in our electronics and home automation projects, we can use the SCT013 current sensors to measure the electrical consumption of a device, check the status of an electrical installation, and to record the consumption of electricity in home energy monitors. an installation or even access through the internet in real time.

The most common model is SCT013-000, of which the maximum current is 100A, the current output is 50mA (100A:50mA), the maximum current of SCT-013-030 is 30A (30A/1V), and the voltage output is 1V.

Finally, while it is important to have a wide range of measurements, it is important to bear in mind that a higher intensity model will result in less precision. An intensity of 30A to 230V corresponds to a load of 6,900W, which is enough for most home users.

How does the SCT013 work?

The SCT013 sensors are small current transformers, or CTs. Current transformers are instruments widely used for measuring elements.

A current transformer is similar to a voltage transformer and is based on the same operating principles (in fact, they were previously identical). However, they have different objectives and, as a result, are designed and constructed differently.

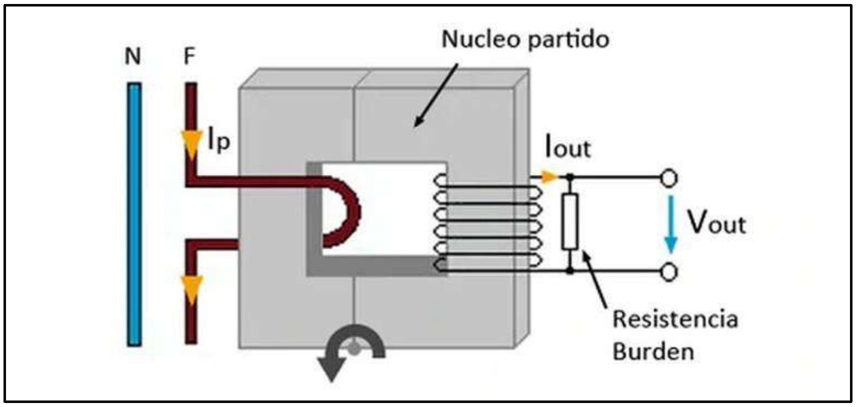


Figure 4.9: Current Transformer

A current transformer seeks to generate an intensity in the secondary that is proportional to the intensity that passes through the primary. For this, it is desired that the primary is formed by a reduced number of turns.

We can use the current transformer to build non-intrusive current sensors. In the current sensor, the ferromagnetic core can be separated so that the conductor can be opened and rolled up.

So, we have a transformer, it is:

- The cable through which the current to be measured is the primary winding
- The “clamp” is the magnetic core.
- The secondary winding is integrated as part of the probe.

When the alternating current circulates through the conductor, a magnetic flux is generated in the ferromagnetic core, which in turn generates an electrical current in the secondary winding.

The intensity transformation ratio depends on the relationship between the number of turns:

$$\boxed{\frac{I_s}{I_p} = \frac{Vp}{Vs} = \frac{Np}{Ns}}$$

The primary is usually formed by a single loop made by the conductor to be measured. Although, it is possible to wind the driver making this happen more than once inside the “clamp”. The number of turns of the secondary, integrated in the probe, varies from 1000-2000, according to the models of the SCT013.

Unlike voltage transformers, in a current transformer, the secondary circuit should never be opened, because induced currents could damage the component. For this reason, the sensors of SCT13 have protections: resistance burden in the sensors of output by voltage, or diodes of protection in the sensors of exit by the current.

To understand the connection of the sensor SCT013, we have to understand and solve three problems:

- Sensor output in intensity
- Adjustment of the voltage range
- Positive and negative stress

SENSOR OUTPUT IN INTENSITY

The SCT013 are current transformers, that is, the measurement is obtained as an intensity signal proportional to the current flowing through the cable,. Processors, however, are only capable of measuring voltages.

This problem is easy to solve. To convert the output in intensity into a voltage output, we only have to include a resistance (load resistance).

With the exception of model SCT013-100, all other SCT013 models have an internal load resistance so that the output is a voltage signal of 1V. This is why it will not be a concern to worry about.

Only in the case of SCT013-100, there is no resistance internal burden, so the output is a signal of ± 50mA. A resistance of 33Ω in parallel with the sensor will suffice.

Positive and Negative Tensions

Another problem we have to solve is that we are measuring alternating current, and the intensity induced in the secondary is alternating. After passing through the resistance burden, whether internal or external, the voltage output is also alternating.

However, as we know, the analog inputs of the majority of processed currents, including Arduino, can only measure positive voltages.

To measure the voltage at the transformer output, we have several options, ordered here from least to most recommended:

1. Rectify the signal through a diode bridge, and measure the wave as positive values. Not advisable given that we lose information as to whether we are in the negative or positive half-period, and because we will have the voltage drop of the diode, and, even worse, the diode does not drive below a voltage meaning the signal will be distorted at the junctions by zero.

2. Add an offset in DC by using two resistors and a capacitor that provide a midpoint between GND and Vcc. Much better if we also add an operational amplifier as a voltage follower.

3. Add an ADC with differential input, which allows measurements of positive and negative voltages, such as the ADS1115. This is the option that we are going to use.

Voltage Range Adaptation

The last problem is the need to adapt the range of voltages at the sensor output. Arduino can only perform measurements between 0 and Vcc. In addition, the smaller the range, the greater loss of accuracy, so we should adapt to this range.

On the other hand, we must remember that when it comes to AC voltage, RMS values are usually used. Briefly review the peak voltage and peak-to-peak equations:

$$\boxed{V_{pico} = \sqrt{2} \cdot V_{RMS} = R \cdot I_{pico} = \sqrt{2} \cdot R \cdot I_{RMS}}$$

Therefore, for a sensor with an output of ±1V RMS, the peak voltage is ±1.414V and the

In the case of the SCT013-100, the output will be ±50mA. With an external load resistance of 33Ω, the output voltage is ±1.65V RMS, so the peak voltage is ±2.33V and the peak-to-peak voltage is 4.66V.

4.4 Display Utility

- How TFT LCD Work

When compared to the ordinary LCD, TFT LCD gives a very sharp and crisp picture/text with a shorter response time. TFT LCD displays are used in more and more applications, giving products better visual presentation.

- Structure of TFT LCD

TFT is an abbreviation for "Thin Film Transistor". The color TFT LCD display has transistors made up of thin films of Amorphous silicon deposited on a glass. It serves as a control valve to provide an appropriate voltage onto liquid crystals for individual sub-pixels. That is why the TFT LCD display is also called Active-Matrix display.

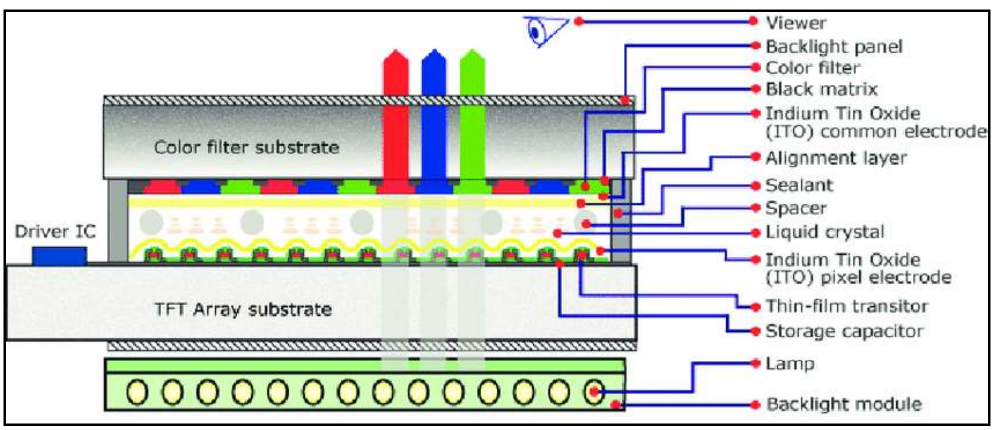


Figure 4.10: Structure of TFT LCD

A TFT LCD has a liquid crystal layer between a glass substrate formed with TFTs and transparent pixel electrodes and another glass substrate with a color filter (RGB) and transparent counter electrodes. Each pixel in an active matrix is paired with a transistor that includes a capacitor which gives each sub-pixel the ability to retain its charge, instead of requiring an electrical charge sent each time it needs to be changed. This means that TFT LCD displays are more responsive. **[23]**

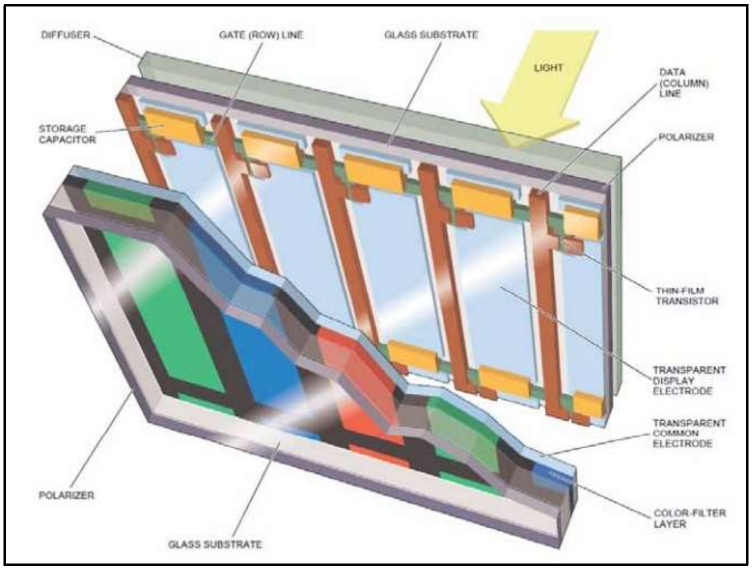


Figure 4.11: Structure of the TFT LCD

- **Configuration on RP3**

Name	Description	Rp0
LED	Led on, off	3.3v
SCK	SPI Clock	Pin 21
SDA	Data (MOSI)	Pin 20
A0 or DC	(0) Command, (1) Data	Pin 24
RESET	For reset	Pin 25
CS	Chip select	Pin 18
GND	Ground	Ground
VCC	Power on, off	3.3v

Table 4.1: Configuration on RP3

Displays on RP3 version:

1- Sponsor logo elsewedy electrical iskraemeco.

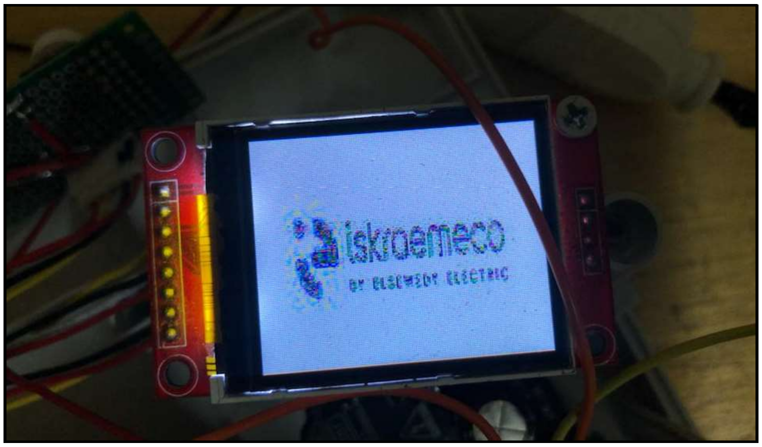


Figure 4.12: Iskraemeco

2- Team logo /Root

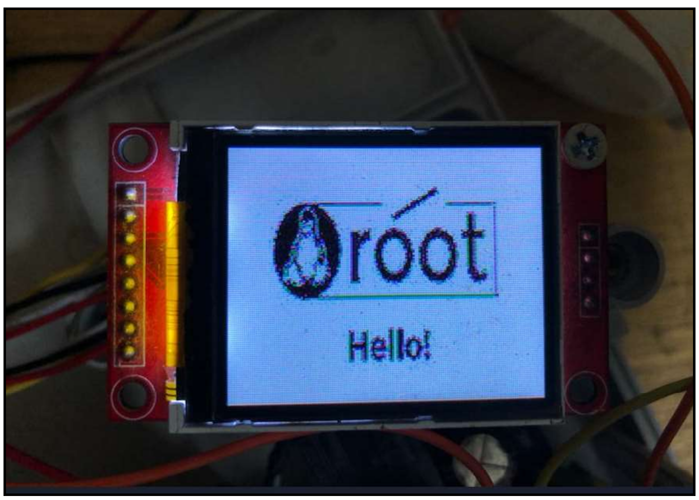


Figure 4.13: /root

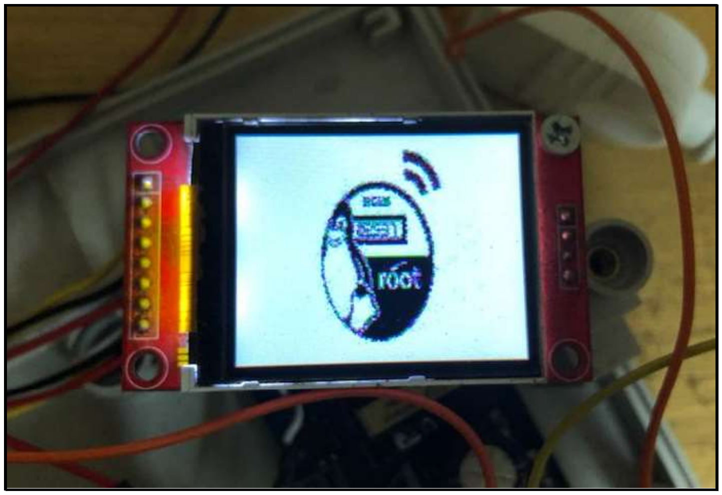


Figure 4.14: /root NGSM

3- Voltage, current, energy, power.

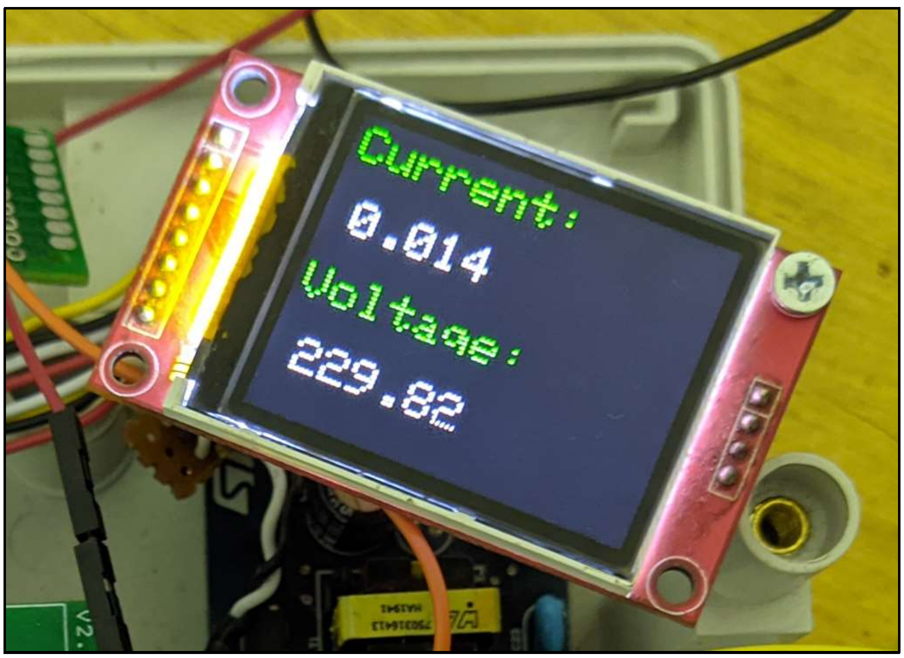


Figure 4.15: Electrical Data 1

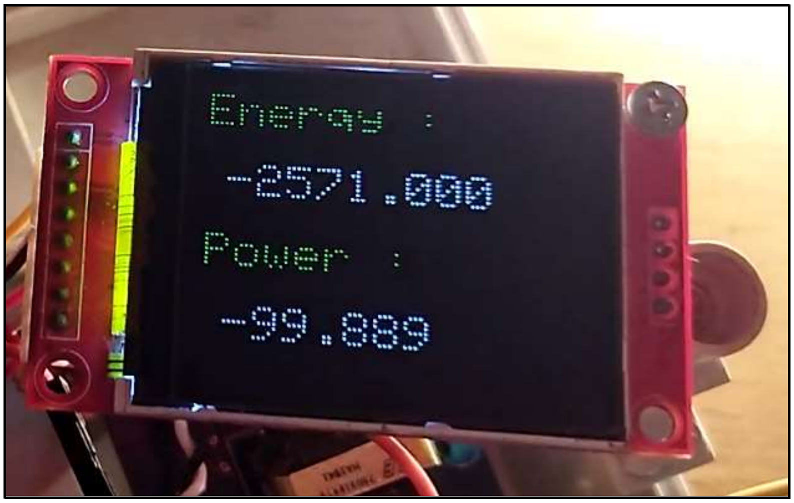


Figure 4.16: Electrical Data 2

4.5 Connection to the Grid

Meters send readings to the server on the cloud for further calculations like calculate the debt(money) of energy consumption and also to store the measurements also the meter sends the events like tampering to the cloud to informs the utility.

Results Messages

Search to filter items...

id	active_power	amount	date	electric_current	energy	re_active_power	volt	meter_id
601		1100	2022-07-14T03:40:32.0000000	0	100		0.1	43
602		1650	2022-07-13T03:40:32.0000000	0	150		0.3	43
603		2970	2022-07-12T03:40:32.0000000	0	270		0.3	43
604		5390	2022-07-12T03:40:32.0000000	0	490		0.3	43
605		6270	2022-07-12T03:40:32.0000000	0	570		0.3	43
606		6710	2022-07-11T03:40:32.0000000	0	610		0.3	43
607		7040	2022-07-10T03:40:32.0000000	0	640		0.3	43

Figure 4.17: NGSM Backend Server Database

4.6 Procedure of Operation and Technical Demo

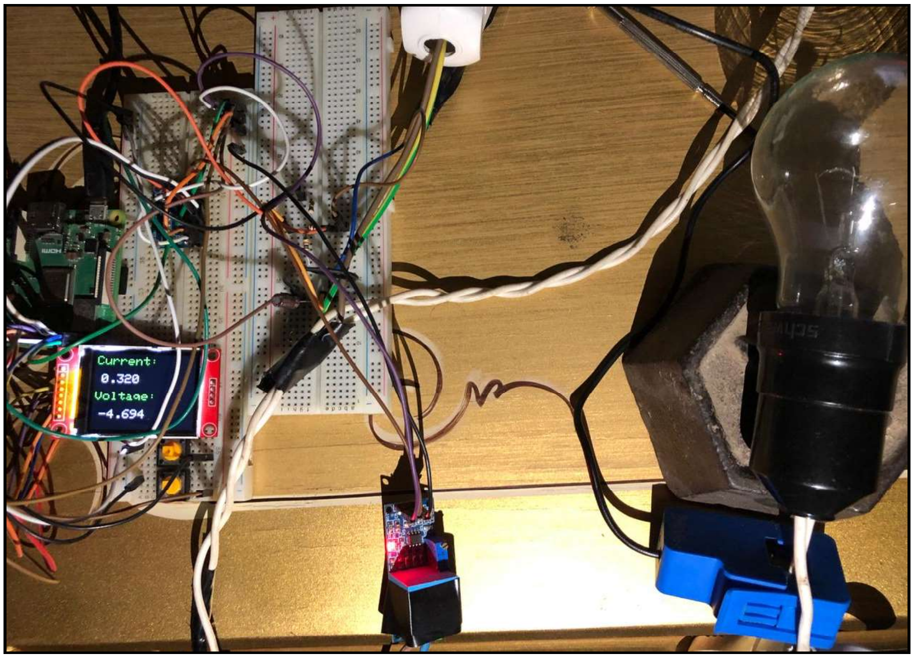


Figure 4.18: Measuring without Input

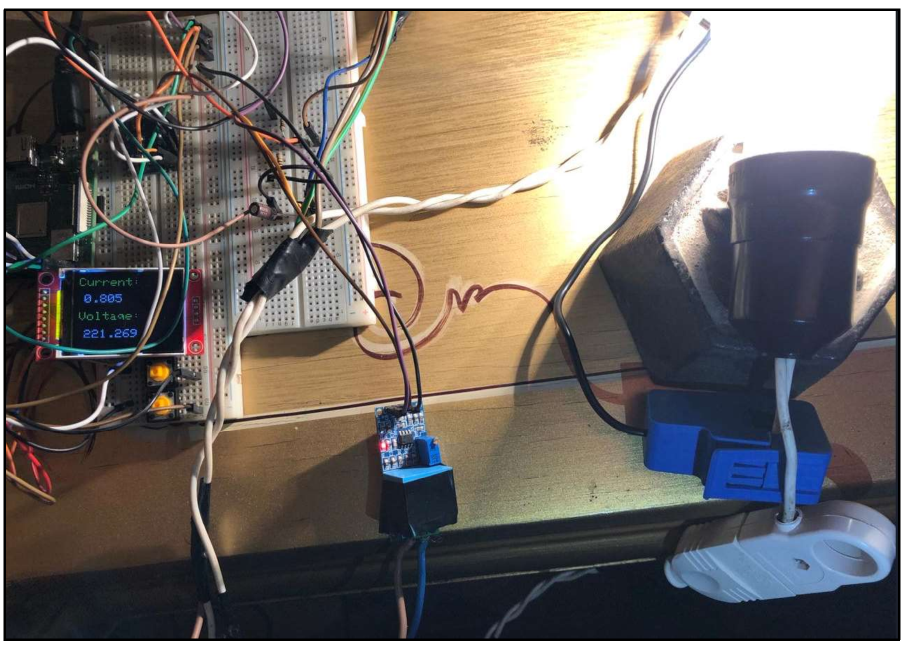


Figure 4.19: Measuring with Load

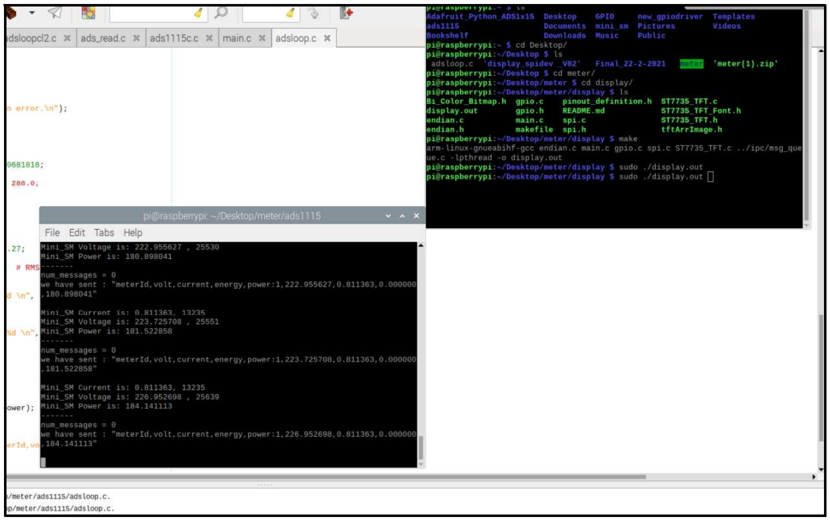


Figure 4.20: Uploading the readings and the events to Backend

4.7 Cost and Bill of Material

S.N	NAME	PURPOSE	Quantity	PRICE
1.	Raspberry Pi 3	Development board	1	15$
2.	ADC	Analog-to-digital Converter	1	3.6$
3.	Voltage Sensor	AC Voltage Sensor	1	
4.	Current Sensor	AC Current transformer	1	2.5$
5.	LCD Display	Display Results	1	8$
6.	Resistor	2W 10K	1	<1$
7.	Resistor	0.5W 10K	2	<1$
8.	Resistor	33 ohm	1	<1$

9.	Resistor	2W 3.3K	1	<1$
10.	Capacitor	50V 10uF	2	<1$
11.	Capacitor	50V 1uF	1	<1$
12.	Diode	IN4007	4	<1$
13.	Diode	3.6V Zener	1	<1$
14.	Potentiometer	10K	1	<1$
15.	Connecting Wires	Jumper Wires	10	<1$
16.	BreadBoard	-	1	<1$
			TOTAL	32$

Table 4.2: Cost and bill of material of 1st version

4.8 Performance Evaluation

The 1st generation of our system was not accurate enough due to the precision and calibration limits of the available sensors. So we enhanced this system to the 2nd generation NGSM. The 2nd generation has more accurate readings as it introduces a powerful sensing unit (AFE) that reads a high frequency to get more features for better appliance detection. Also used Raspberry PI Zero W to be more optimized with better power consumption for continuous usage as the real life electricity meter.

5. 2nd Generation NGSM Smart Meter

5.1 Hardware Specifications

As 2nd generation is enhanced for the 1st we replaced the main SoC from Raspberry-Pi 3 to Raspberry-Pi Zero as it has much lower cost and sufficient processing power for running our Software Implementation. Raspberry-Pi zero has more than one model available in the market. We selected the best which would fit for our implementation which was Raspberry-Pi zero W. It is a single core clocked at 1GHz, supporting arm11 architecture with 512MB DDR2 and also supporting Wi-Fi & Bluetooth.

We used STPM33 as an AFE unit, it supports UART and SPI, we selected SPI as it's faster to communicate with STPM33.

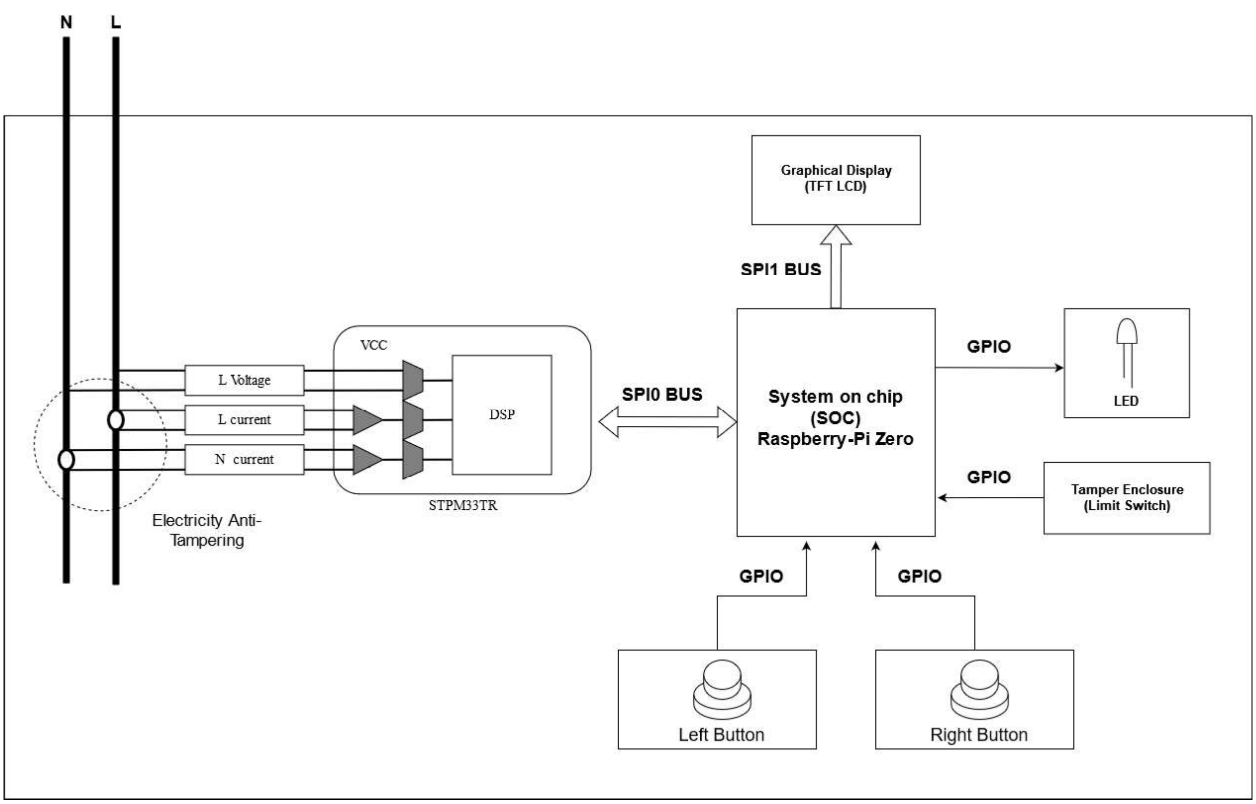


Figure 5.1: 2nd Generation NGSM Hardware Components

As an evolution of the 1st version, we used STPM33 as Analog Front End (AFE) due to its capabilities and as it consists of a DSP unit which provides the ready-measured parameters. We also moved to Raspberry-Pi Zero Model W as it has on board Wi-Fi to connect on the network. It is also a single core which permits optimisation in the power consumption unlike Raspberry-Pi Zero 2W which has 4 Cores and Raspberry-Pi 3 B+ which is used in the previous version. **[24]**

The following figure shows the schematic on Altium designer.

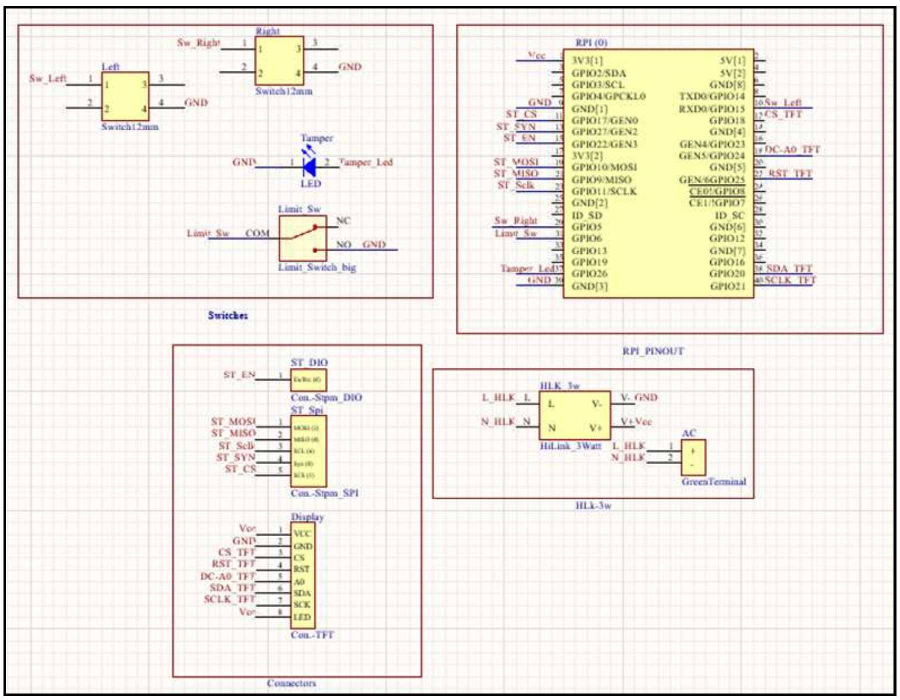


Figure 5.2: PCB Schematic

The Layout is double Layer with some vias to allow flexibility in placement as shown in the following figure.

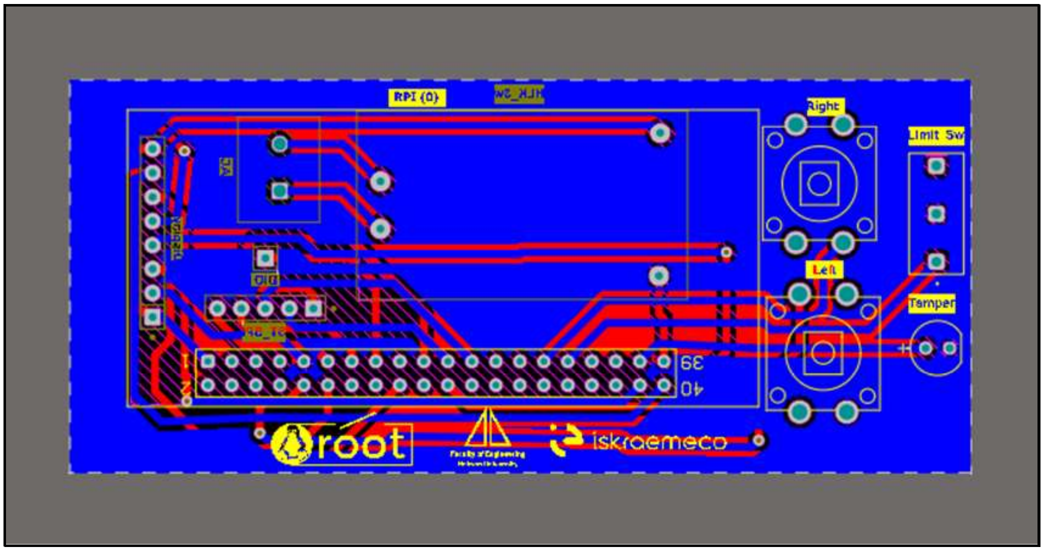


Figure 5.3: PCB Layout

The final 3D model of the PCB on Altium Designer is shown in the following figure.

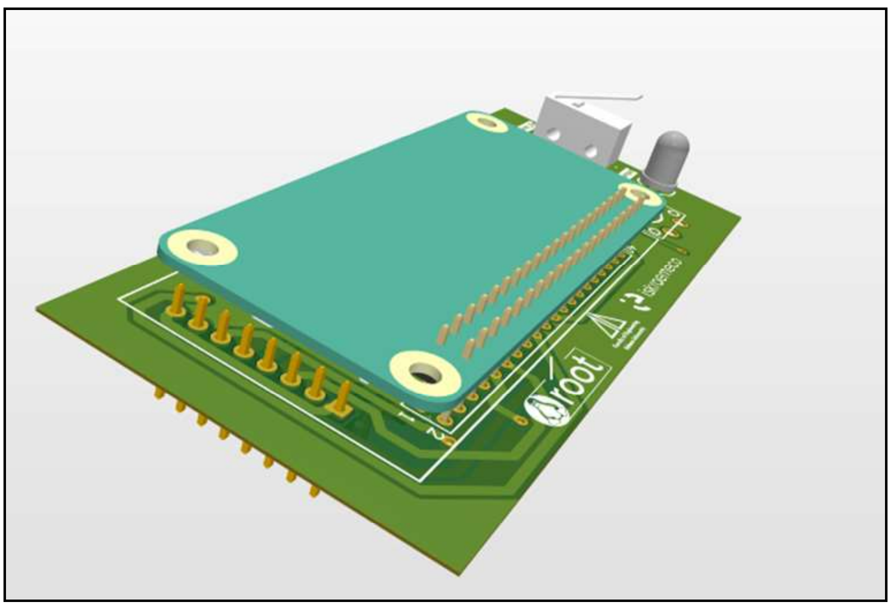


Figure 5.4: PCB 3D Model

5.2 Software Implementation

As we are dealing with POSIX in Linux that makes our software portable to run from architecture to another that has the ability to run Linux. So, we can port our written software on the Raspberry-Pi without any problems. The main differences are hardware dependencies for example spi pins will differ from SoC to another, the procedure of operation and so on.

The use of spidev in userspace

SPI devices have a limited userspace API, supporting basic half-duplex

read() and write() access to SPI slave devices. Using ioctl() requests,

full duplex transfers and device I/O configuration are also available.

```
#include <fcntl.h>

#include <unistd.h>

#include <sys/ioctl.h>

#include <linux/types.h>

#include <linux/spi/spidev.h>
```

Some reasons you might want to use this programming interface include:

* Prototyping in an environment that's not crash-prone; stray pointers

  in userspace won't normally bring down any Linux system.

* Developing simple protocols used to talk to microcontrollers acting

  as SPI slaves, which you may need to change quite often.

Of course, there are drivers that can never be written in user space, because

they need to access kernel interfaces (such as IRQ handlers or other layers of the driver stack) that are not accessible to user space.

Interfacing Raspberry-Pi Zero with STPM33 & TFT requires spidev to be enabled

To enable spidevx.x in Raspberry-Pi

1-Open config.txt in /boot

```
$ sudo nano /boot/config.txt
```

2-Add the following to config.txt

```
dtoverlay=spi0-0cs

dtoverlay=spi0-1cs

dtoverlay=spi1-0cs

dtoverlay=spi1-1cs
```

3-Then save the file using

ctrl+shift x

y

enter

4-Reboot your system using

```
$ sudo reboot
```

5-To check out write the following command

```
$ sudo ls -l /dev/spi*
```

The output will be similar to

```
crw-rw---- 1 root spi 153, 1 Feb 23 13:17 /dev/spidev0.0

crw-rw---- 1 root spi 153, 2 Feb 23 13:17 /dev/spidev0.1

crw-rw---- 1 root spi 153, 0 Feb 23 13:17 /dev/spidev1.0

crw-rw---- 1 root spi 153, 0 Feb 23 13:17 /dev/spidev1.1
```

Note that /dev/spidev0.0 & /dev/spidev0.1 are on the same spi bus but with different chip select.

Interfacing Raspberry-Pi with TFT LCD

As we have multiple pages will be shown on the LCD

We used **pthreads** in TFT Module as regular forever loop won't fit

regular embedded code is run in forever loop such as that

```
start:

//some code

loop:

//some code

forever;
```

When using multi-threading there will be more than one forever loop each runs as in parallel state using scheduling algorithms

```
start:

//some code

loop1:

//some code

forever;

loop2:

//some code

forever;

loop3:

//some code

forever;
```

Semaphore

A semaphore is a data structure used to help threads work together without interfering with each other. The POSIX standard specifies an interface for semaphores; it is not part of Pthreads, but most UNIXes that implement Pthreads also provide semaphores. Semaphores provide synchronization between the multiple threads and it also prevents race conditions for shared resources.

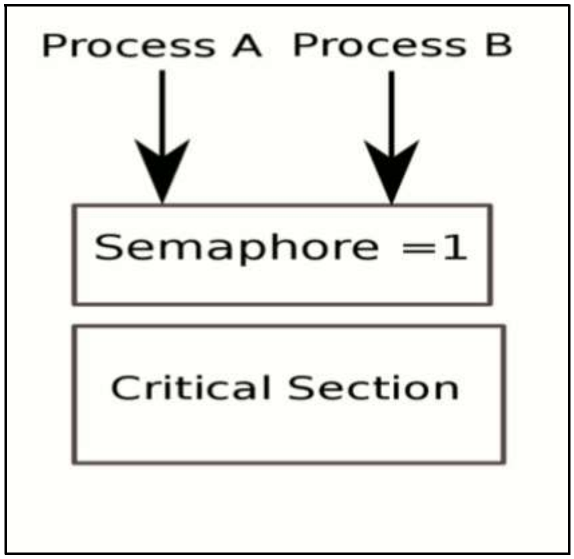


Figure 5.5: Binary Semaphore illustration

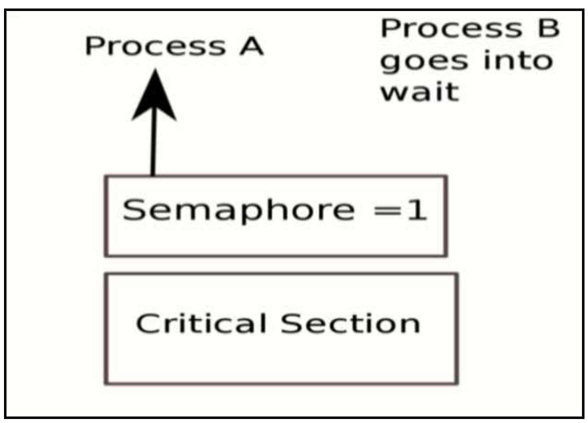


Figure 5.6: Process B is blocked until Process A releases the Semaphore

Interfacing Raspberry-Pi with Limit Switch

Limit switch is working as a simple push button once the limit switch is pressed the default state will be changed and read by the events programs and transferred to Database with the timestamp.

Installing CURL

As Raspberry-Pi has on board Wi-Fi, you can connect to the internet and use the apt command to install whatever you want.

To install the curl

```
$ sudo apt-get install libcurl4-openssl-dev
```

How to run the programs as daemons?

As demonstrated before the daemon is such a process which runs in the background. To make the process run as a daemon, we have to fork it from the parent process.

```
pid_t pid;

pid = fork(); //Fork the Parent Process

if (pid < 0) { exit(EXIT_FAILURE); }
```

5.3 Electricity Metering Procedure

Current sensing is performed by 2CTs, while voltage is sensed by voltage dividers. EVALSTPM33 has two currents and one voltage channel.
Main design parameters are listed in the below table.

channel	parameter	Value
Primary current	Current sensor sensitivity	2.4 mV/A
Secondary current		
Primary Voltage	R1 Voltage divider resistor	810 k ohm
	R2 Voltage divider resistor	470 ohm
Secondary Voltage	R1 Voltage divider resistor	810 k ohm
	R2 Voltage divider resistor	470 ohm
any	Cp constant Pulse	41600 pulses/kwh

Table 5.1: Evaluation Board Parameter

A 22 nF capacitor in parallel with R2 implements an antialiasing filter for voltage signal. Regarding the current signal, the antialiasing filter is given by a 10 nF capacitor and 1 kOhm resistors.

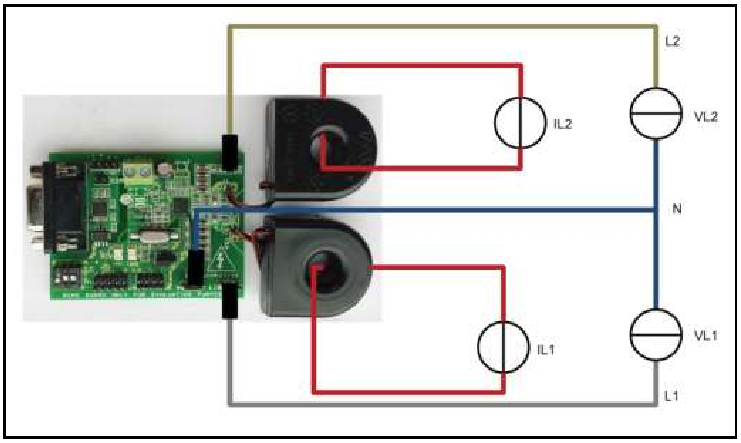


Figure 5.7: Board connection to phantom load, dual-phase system

5.4 Display Utility

- **How TFT LCD Work**

When compared to the ordinary LCD, TFT LCD gives very sharp and crisp picture/text with shorter response time. TFT LCD displays are used in more and more applications, giving products better visual presentation.

- **Structure of TFT LCD**

TFT is an abbreviation for "Thin Film Transistor". The color TFT LCD display has transistors made up of thin films of Amorphous silicon deposited on a glass. It serves as a control valve to provide an appropriate voltage onto liquid crystals for individual sub-pixels. That is why the TFT LCD display is also called Active-Matrix display.

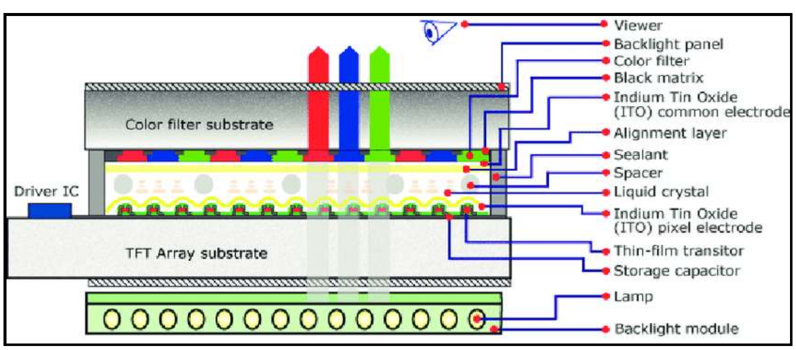


Figure 5.8: Structure of TFT LCD

A TFT LCD has a liquid crystal layer between a glass substrate formed with TFTs and transparent pixel electrodes and another glass substrate with a color filter (RGB) and transparent counter electrodes. Each pixel in an active matrix is paired with a transistor that includes a capacitor which gives each sub-pixel the ability to retain its charge, instead of requiring an electrical charge sent each time it needs to be changed. This means that TFT LCD displays are more responsive.

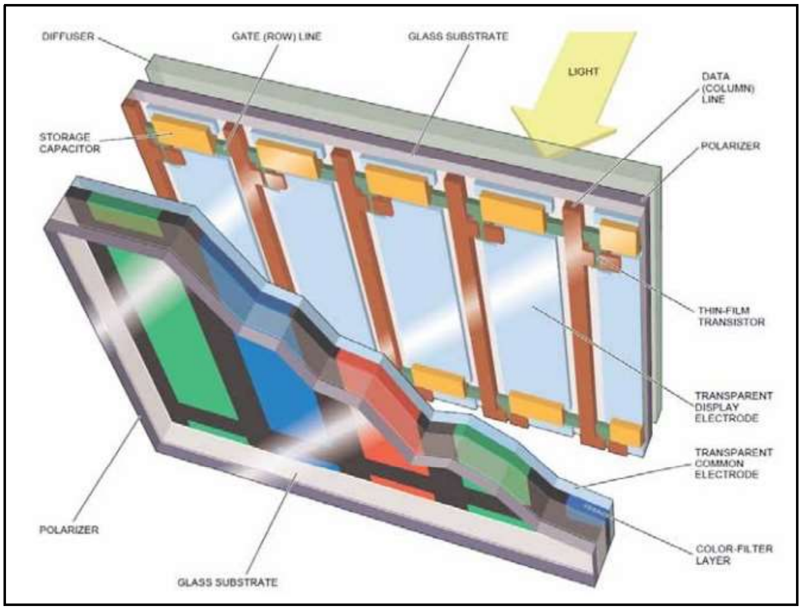


Figure 5.9: Structure of the TFT LCD

Configuration on RP0

Name	Description	Rp0
LED	Led on, off	3.3v
SCK	SPI Clock	Pin 21
SDA	Data (MOSI)	Pin 20
A0 or DC	(0) Command, (1) Data	Pin 24

RESET	For reset	Pin 25
CS	Chip select	Pin 18
GND	Ground	Ground
VCC	Power on, off	3.3v

Table 5.2: Configuration on RP0

5.5 Connection to the Grid

Meters send readings to the server on the cloud for further calculations like calculating the debt(money) of energy consumption and also to store the measurements also the meter sends the events like tampering to the cloud to inform the utility.

Results Messages

Search to filter items...

id	active_power	amount	date	electric_current	energy	re_active_power	volt	meter_ic
601		1100	2022-07-14T03:40:32.0000000	0	100		0.1	43
602		1650	2022-07-13T03:40:32.0000000	0	150		0.3	43
603		2970	2022-07-12T03:40:32.0000000	0	270		0.3	43
604		5390	2022-07-12T03:40:32.0000000	0	490		0.3	43
605		6270	2022-07-12T03:40:32.0000000	0	570		0.3	43
606		6710	2022-07-11T03:40:32.0000000	0	610		0.3	43
607		7040	2022-07-10T03:40:32.0000000	0	640		0.3	43

Figure 5.10: NGSM Backend Server Database

5.6 Procedure of Operation and Technical Demo

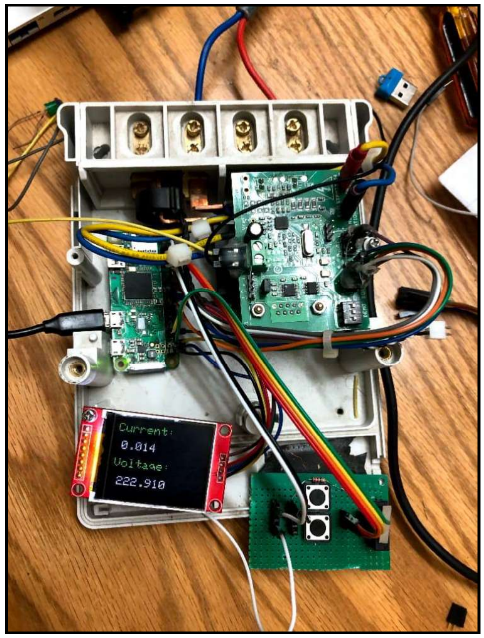


Figure 5.11: Measuring without Input

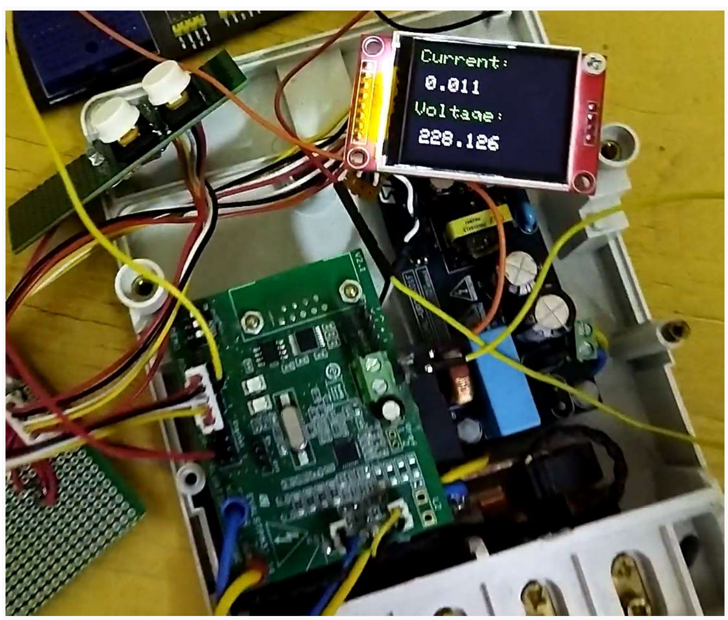


Figure 5.12: Measuring with 220V AC Input

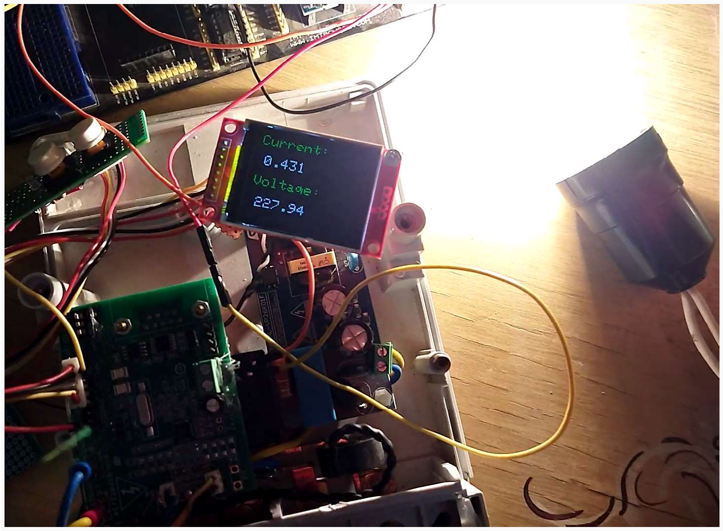


Figure 5.13: Measuring with Load

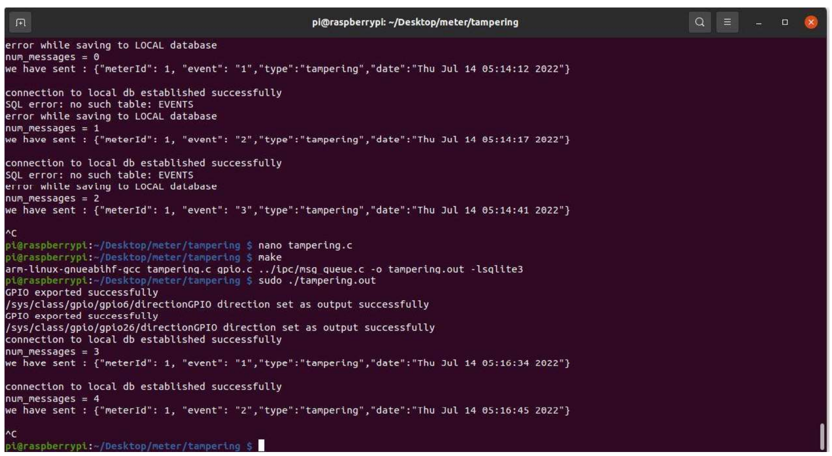


Figure 5.14: Enclosure Tampering Detection

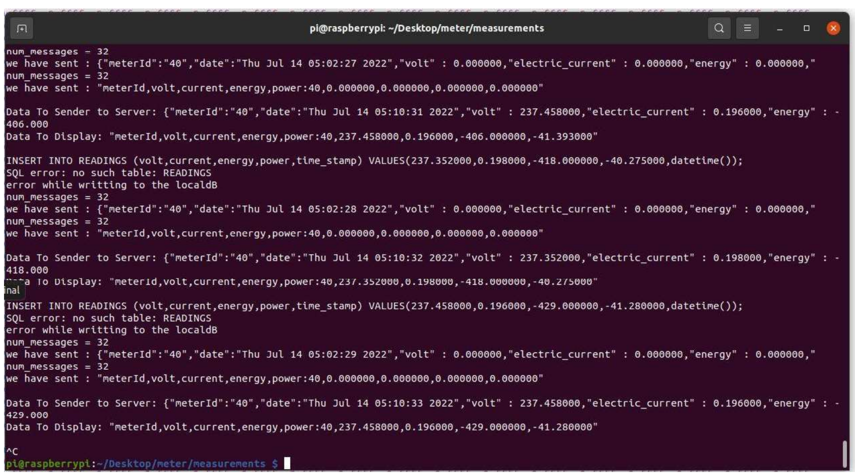


Figure 5.15: Uploading the readings and the events to Backend

5.7 Cost and Bill of Material

Cost and bill of material of 2nd version

Item No.	Item name	Specification	Quantity	Price unit	Purpose of item
1	Raspberry Pi Zero 2 W	https://www.raspberrypi.com/products/raspberry-pi-zero-2-w/	1	$15	We build the whole system on it , and through it we can control the system
2	STPM 33TR	https://www.st.com/en/data-converters/stpm33.html	1	$74.8	It is the measuring unit in the system , it has the ability to calculate the current and volt
3	Hi-Link HLK-PM03 AC/DC Converter Module (3.3V)	https://www.electronicscomp.com/hlk-pm03-hi-link-3.3v-3w-ac-dc-power-supply-module	1	$1.75	It supply the system with the needed power
4	ST77355 TFT Screen	https://www.st.com/resource/en/application_note/an4861-lcdtft-display-contr	1	8$	It display the readings of meter to user

		oller-ltdc-on-stm32-mcus-stmicroelectronics.pdf			
5	Limit switch		1	$0.2	It is used for tampering detection
6	button		2	$0.2	It is used for interface between user and system
7	LED		1	$0.2	It is used for tampering detection
	Total cost			$100	

Table 5.3: Cost and bill of material of 2nd version

5.8 Performance Evaluation

As we mentioned before RPI zero is clocked at 1GHz which normally consumes more power, it also has a little bit higher cost for production. That's why we moved to the 3rd upgrade which will solve these problems with keeping the same performance.

6. 3rd Generation NGSM Smart Meter

6.1 Hardware Specifications

The 3rd Generation comes with enhancement in Power consumption and the cost that why we replaced the main SoC from Raspberry-Pi Zero to F1C200S it has much lower cost and lower power consumption than Raspberry-Pi Zero. but the SoC was having some issues and limitations in the Hardware specs. One of the Hardware limitations of F1C200S was the availability of one SPI Bus, that seems for the first while there's no problem as SPI is Multi-Slave Bus but when you go deeper you will find that the STPM33 working in Mode3 while the TFT LCD working on Mode0 which causes interference when they on the same Bus even with disabling the Chip Select for so that was required kind of isolation that's why we tried 2 solution such as Demultiplexer and Tri-State Buffer as shown in the following figure.

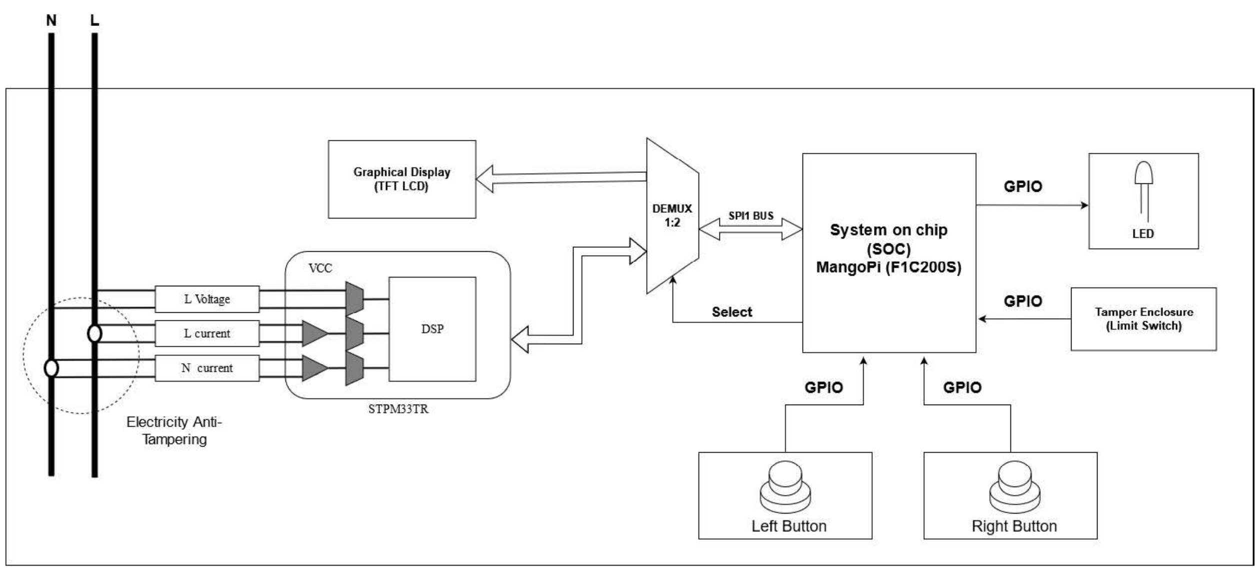


Figure 6.1: 3nd Generation NGSM Hardware Components

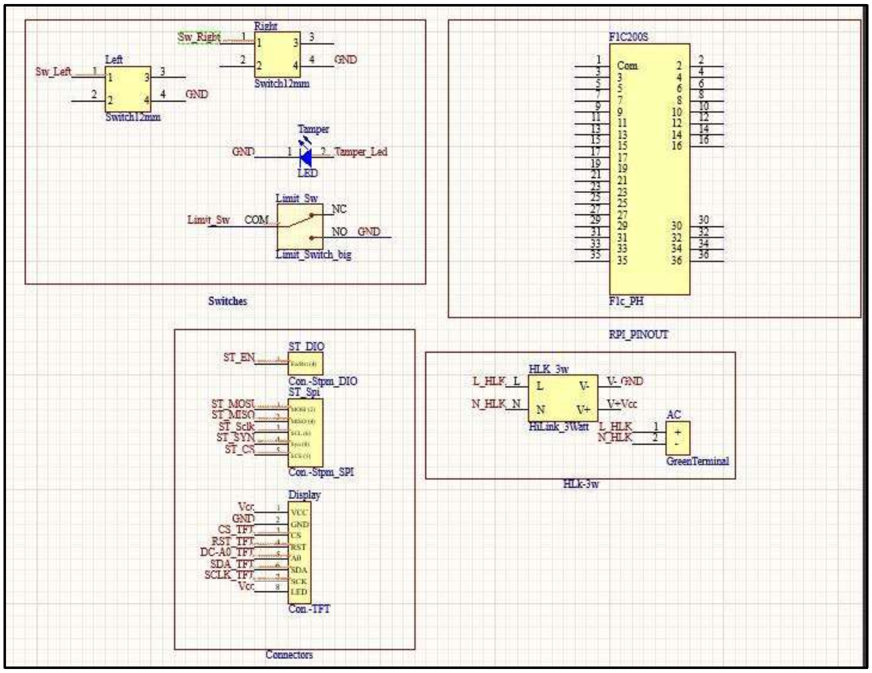


Figure 6.2: PCB Schematic of 3rd Generation

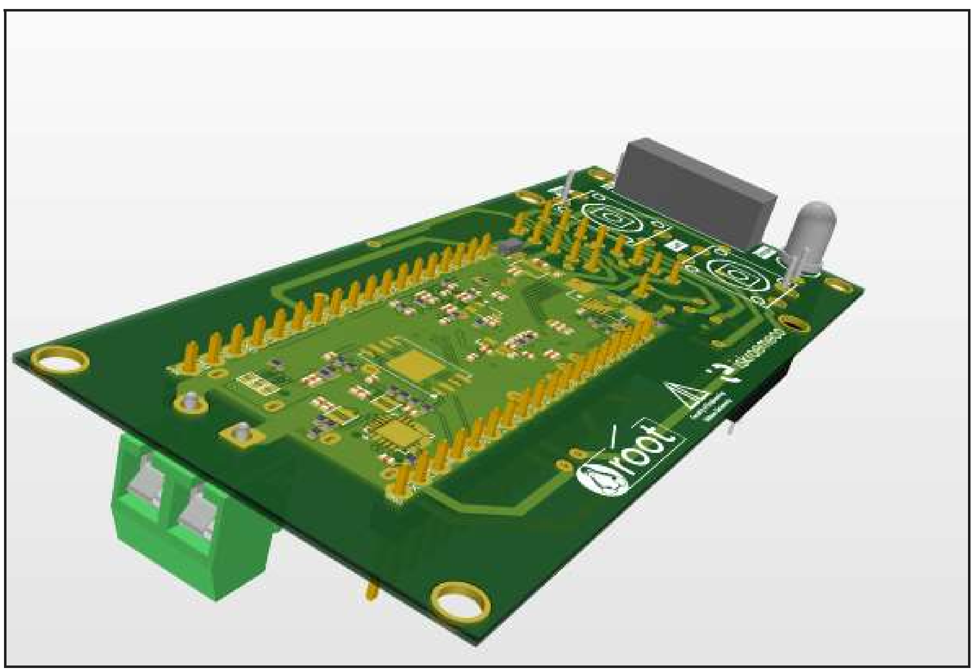


Figure 6.3: PCB 3D Model of 3rd Generation

6.2 Software Implementation

The F1C200S differs from Raspberry-Pi zero as it has very limited Linux functionalities for example you can't use apt commands to install some libraries and of course you can't use native compilation using gcc on it. That's why we had to cross compile all the programs to extract the executable files which can be run on the F1C200s directly.

First of all, we have to build the Linux image using Buildroot, There's supported working Linux image using BuildRoot.

After cloning the BuildRoot files, we have to generate the .config file by the following command

```
$ make widora_mangopi_r3_defconfig
```

buildroot offers graphical terminal to select the required configurations and adds-on for Linux, it's called menuconfig, by typing the following command you will be able to run menuconfig

```
$ make menuconfig
```

The menuconfig let

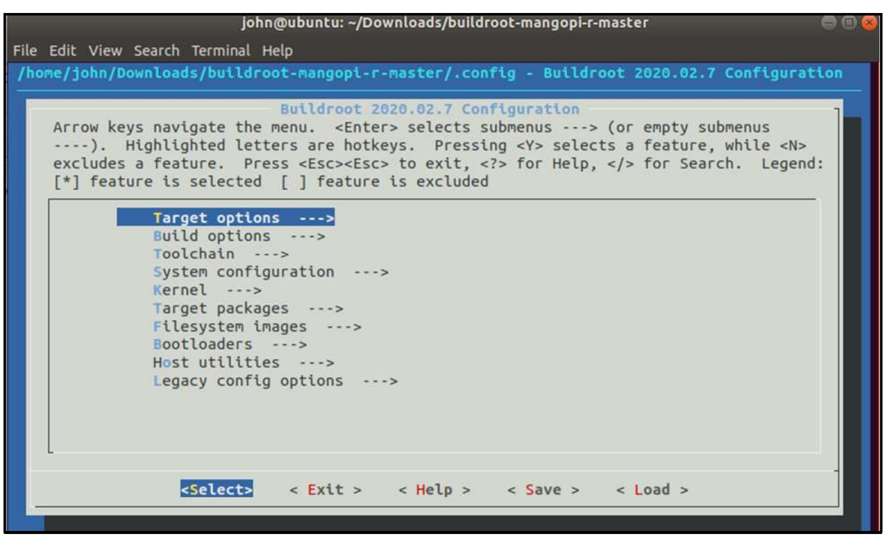


Figure 6.4: buildroot menuconfig

When we explore Target options, we find the Target Architecture & variant which is arm926t little endian supporting soft floating point as it doesn't have FPU as shown in **Fig 6.5**.

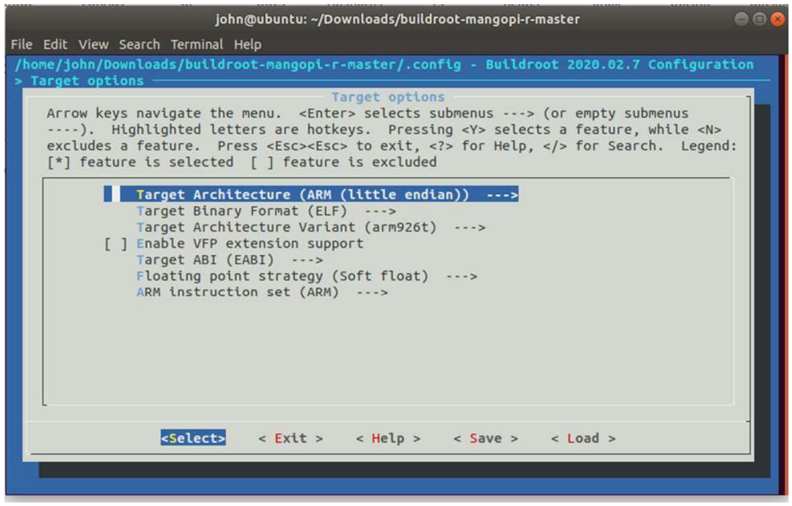


Figure 6.5: Target options in buildroot

And now let's enable the needed functionalities.

For example we can select the enabled text editor to be added to the Linux image as shown in **Fig 6.6**.

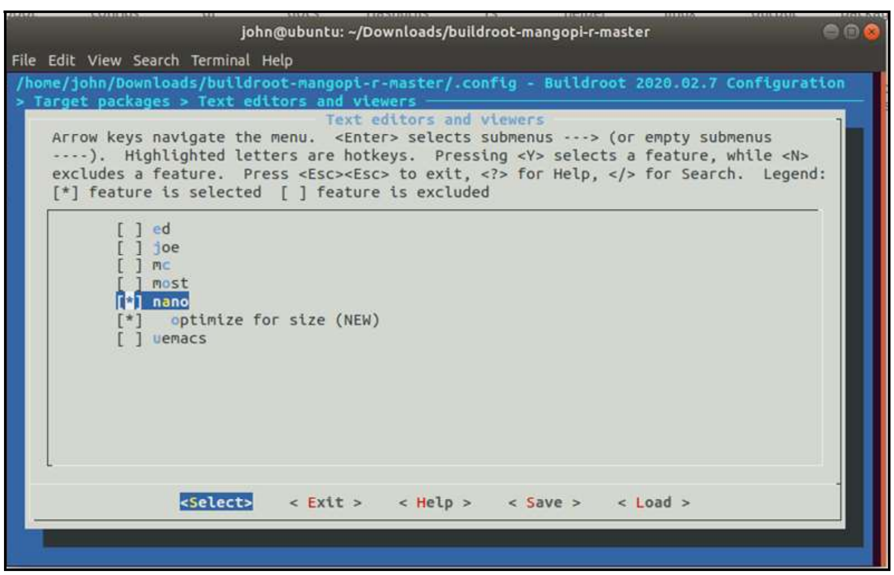


Figure 6.6: Enable nano text editor in buildroot

After selecting the required configurations for your Linux kernel you can build the kernel now by typing the following magically command

```
$ make -j4
```

It will take some quiet time based on your machine performance, you can accelerate the operation by adding -j4 where 4 is the number of cores. The output will be something as shown in **Fig 6.7**.

Figure 6.7: building Linux

After building the image you will find an output directory which contains some directory, we are now with the target directory. This directory contains the sysimage-sdcard.img that will be the image to be burned to the SD Card to boot the image.

Figure 6.8 : the output directory in buildroot

By plugging the SD Card in the Mango-Pi and plugging the USB, you will be able to connect to the chip using the serial port as shown in the following figure.

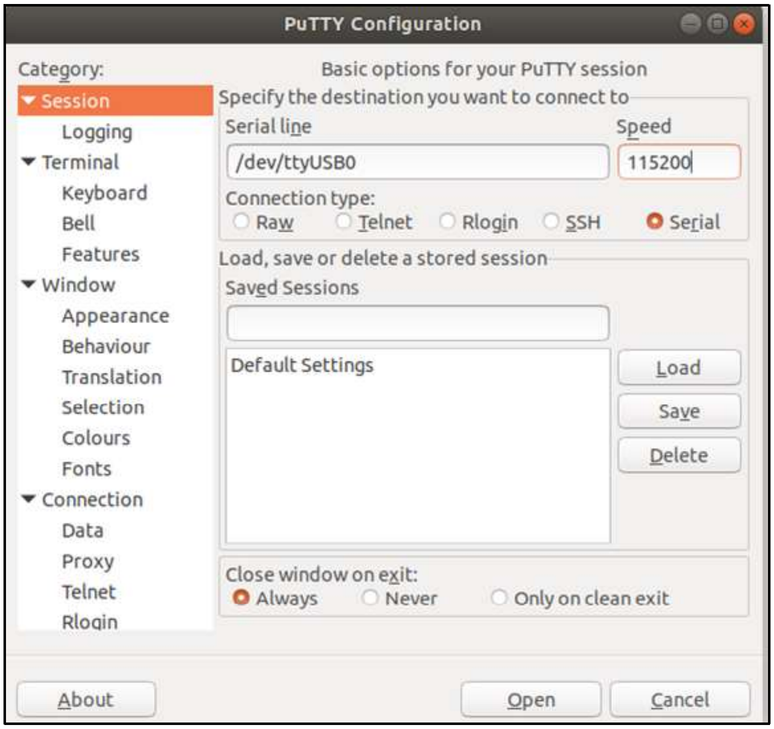


Figure 6.9: connecting using Serial Port in putty

Now you can see the Linux kernel while booting

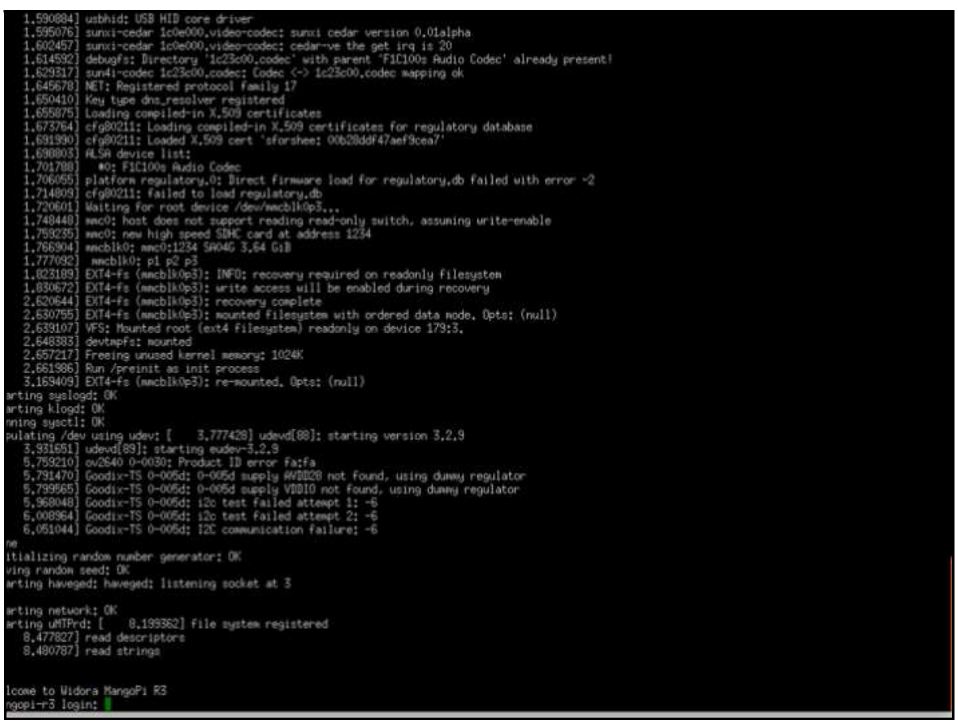


Figure 6.10: Linux booting

To execute simple HelloWorld example, we need to cross compile it

Buildroot doesn't generate only the Linux image but it also generates the cross compiler for the Linux image which you can find in the output directory as shown in **Fig 6.11**.

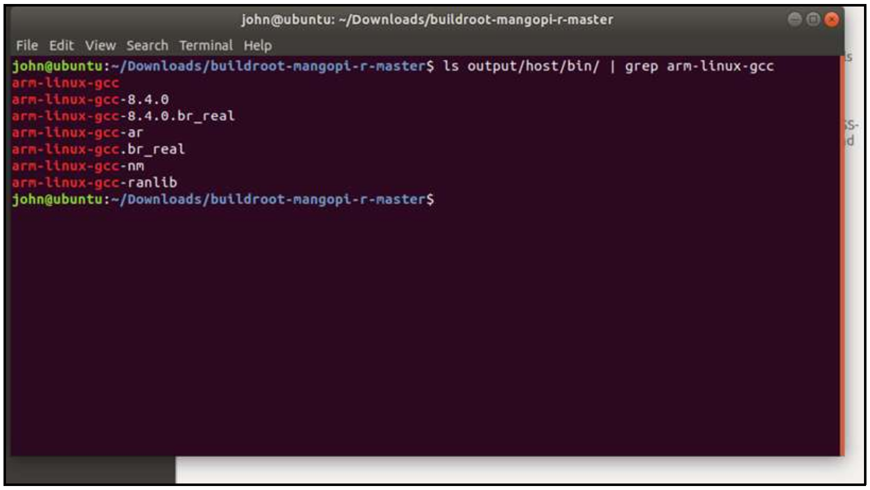


Figure 6.11: arm-linux-gcc cross compiler

Now we can export the PATH of the cross compiler to be able to compile simple Hello World examples as shown in the following figure.

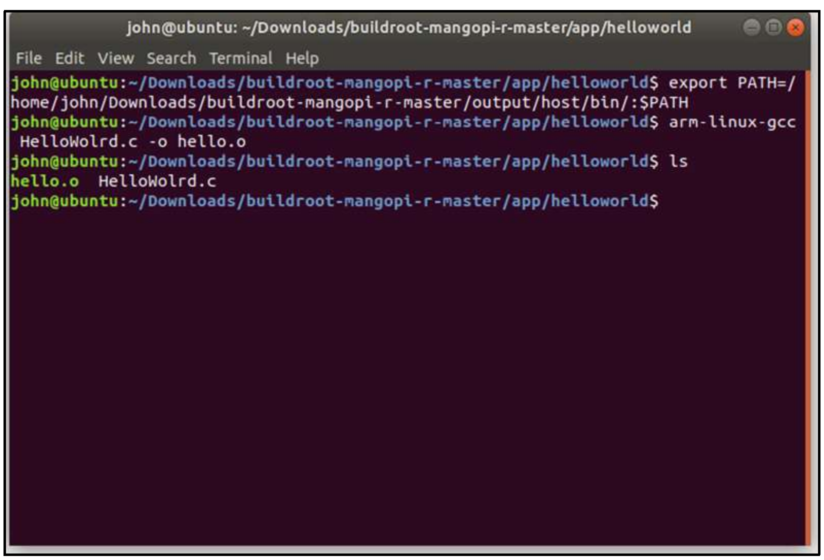


Figure 6.12: cross compilation of simple HelloWorld example

We can move the executable file hello.o to the Mangopi-R3 and execute it as shown in the following figure.

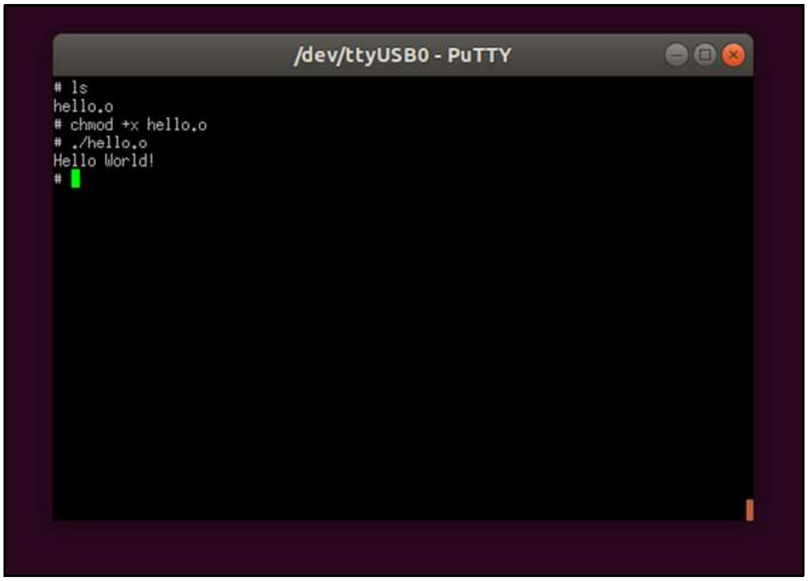


Figure 6.13 : running program on F1C200S

As we have a portable system we can cross compile all the previously discussed software with the same way without any problems but the procedure of enabling the spidev will vary as raspberry pi has config.txt interface but F1C200S has not. So there's another method to enable spidev.c using Linux device tree. **Devicetree** is a data structure describing the hardware components of a particular computer so that the operating system's kernel can use and manage those components, including the CPU or CPUs, the memory, the buses and the integrated peripherals. It has its own syntax which is much similar to C Language but it's not actually.

to enable spidev1.0 & spidev1.1 as follows.

```
&spi1{

  status ="okay";

  spidev@0 {

    status = "okay";

      compatible = "rohm,dh2228fv";

      reg = <0>;
```

```
        spi-max-frequency = <120000000>;

    };

  spidev@1{

    status = "okay";

        compatible = "rohm,dh2228fv";

        reg = <1>;

        spi-max-frequency = <120000000>;

  };

};
```

Then we have to rebuild the kernel using the following:

```
$ ./rebuild-kernel.sh
```

Now when we look for the spidevx.x device in /dev/ we can find them and therefore we can use them in our programs.

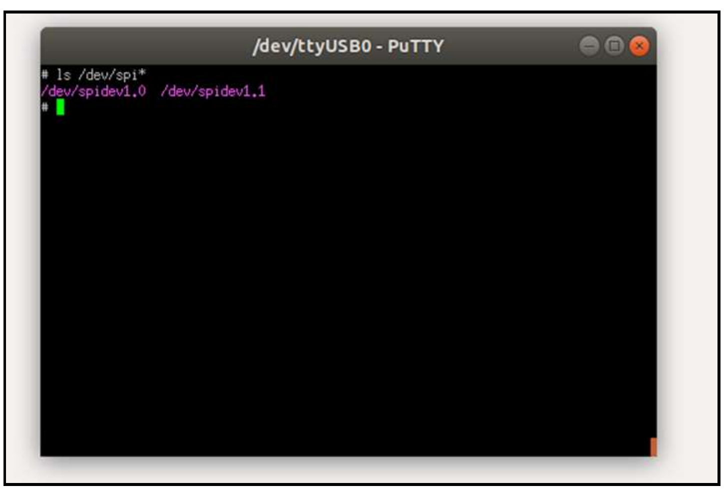


Figure 6.14: Enabled SPI in F1C200S

6.3 Electricity Metering Procedure

Current sensing is performed by 2CTs, while voltage is sensed by voltage dividers. EVALSTPM33 has two currents and one voltage channel.
Main design parameters are listed in the below table. **[25]**

channel	parameter	Value
Primary current	Current sensor sensitivity	2.4 mV/A
Secondary current		
Primary Voltage	R1 Voltage divider resistor	810 k ohm
	R2 Voltage divider resistor	470 ohm
Secondary Voltage	R1 Voltage divider resistor	810 k ohm
	R2 Voltage divider resistor	470 ohm
any	Cp constant Pulse	41600 pulses/kwh

Table 6.1: Evaluation Board Parameter

A 22 nF capacitor in parallel with R2 implements an antialiasing filter for voltage signal. Regarding the current signal, the antialiasing filter is given by a 10 nF capacitor and 1 kOhm resistors.

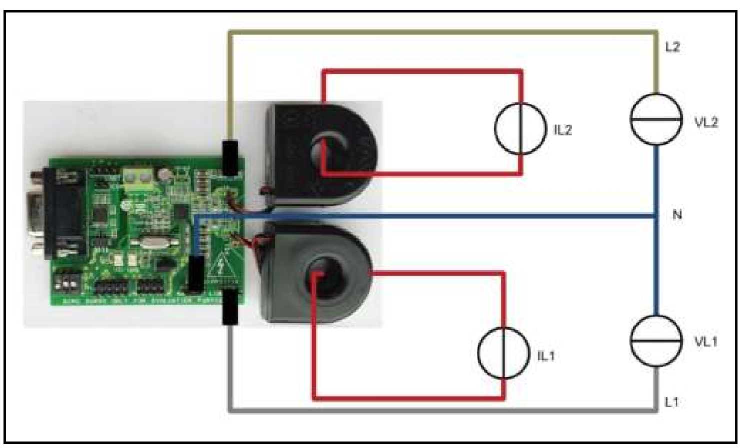


Figure 6.15: Board connection to phantom load, dual-phase system

6.4 Display Utility

Configuration on F1C

Name	Description	To F1C
LED	Led on, off	3.3v
SCK	Clock	Pin 131
SDA	Data (MOSI)	Pin 130
A0 or DC	(0) Command, (1) Data	Pin 132
RESET	For reset	Pin 133
CS	Chip select	Pin 129
GND	Ground	Ground
VCC	Power on, off	3.3v

Table 6.2: Configuration on F1C

6.5 Cost and Bill of Material

Cost and bill of material of 3rd version

Item no.	Item name	Specification	Quantity	Price unit	Purpose of item
1	Allwinner F1C200S	https://linux-sunxi.org/images/1/11/Allwinner_F1C200s_Datasheet_V1.0.pdf	1	$19.6	We build the whole system on it , and through it we

		https://www.cnx-software.com/2020/04/04/widora-tiny200-allwinner-f1c200s-arm9-development-board-support-dvp-camera-up-to-512mb-sd-nand-flash/			can control the system
2	STPM33TR	https://www.st.com/en/data-converters/stpm33.html	1	$74.8	It is the measuring unit in the system , it has the ability to calculate the current and volt
3	Hi-Link HLK-PM03 AC/DC Converter Module (3.3V)	https://www.electronicscomp.com/hlk-pm03-hi-link-3.3v-3w-ac-dc-power-supply-module	1	$1.75	It supply the system with the needed power
4	ST77355 TFT Screen	https://www.st.com/resource/en/application_note/an4861-lcdtft-display-controller-ltdc-on-stm32-mcus-stmicroelectronics.pdf	1	$13	It display the readings of meter to user

5	Limit switch		1	$0.2	It is used for tampering detection
6	button		2	$0.2	It is used for interface between user and system
7	LED		1	$0.2	It is used for tampering detection
	Total			$104.75	

Table 6.3: Cost and bill of material of 3rd version

6.6 Performance Evaluation

F1C200S offers the same performance with optimized power consumption and very lower cost; the F1C200S itself costs less than 4$.

7. NGSM Backend Software System Design (Web-Base) System

7.1 Overview and Summary

In this section, we will discuss the trade-off between the alternative solutions to the problems that faced us like which type of systems to use, Monolithic or Microservices. Which type of database to use, as there is SQL and noSQL. Do we need to share our database and the reason for it, the database replication and the need for tokens and sessions are also important topics to be discussed.

7.2 Backend Software System Design (Web-Base) Block Diagram

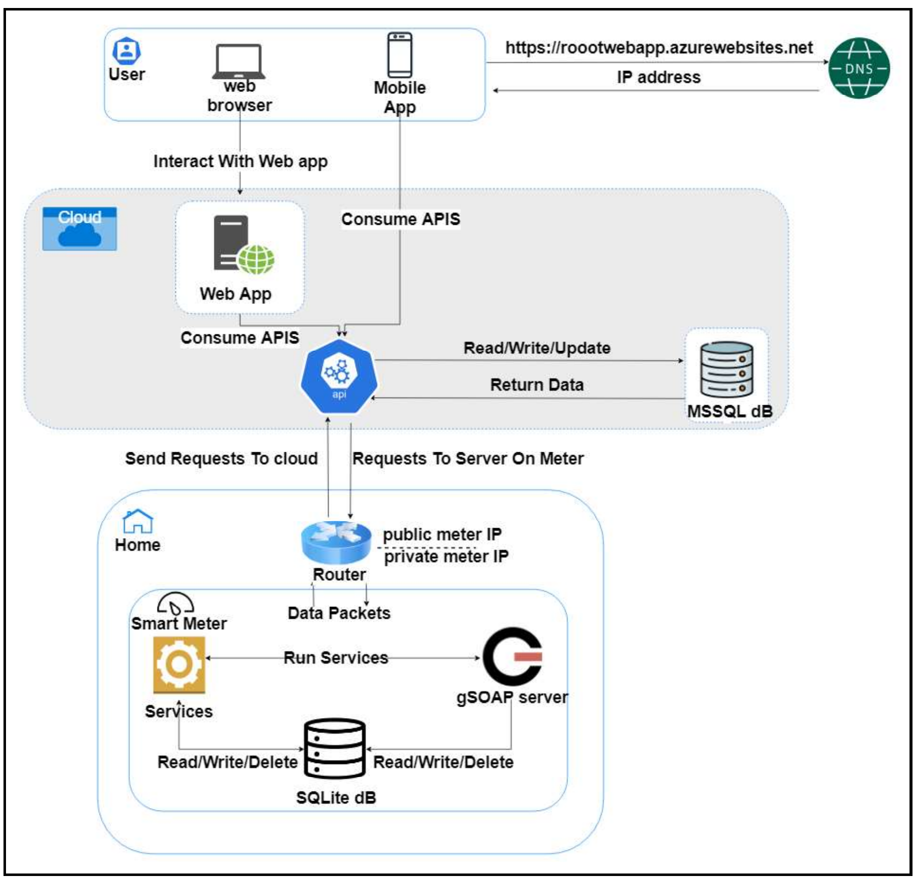


Figure 7.1: NGSM Overall System Architecture

From Figure 7.1, we can see that we have three main blocks:

1- Cloud.

2- Meter.

3- Web Application (which will be included in the cloud part).

We will discuss each block in detail and how each block interacts with the other blocks.

7.3 Main Components and Features

7.3.1 Cloud Software System

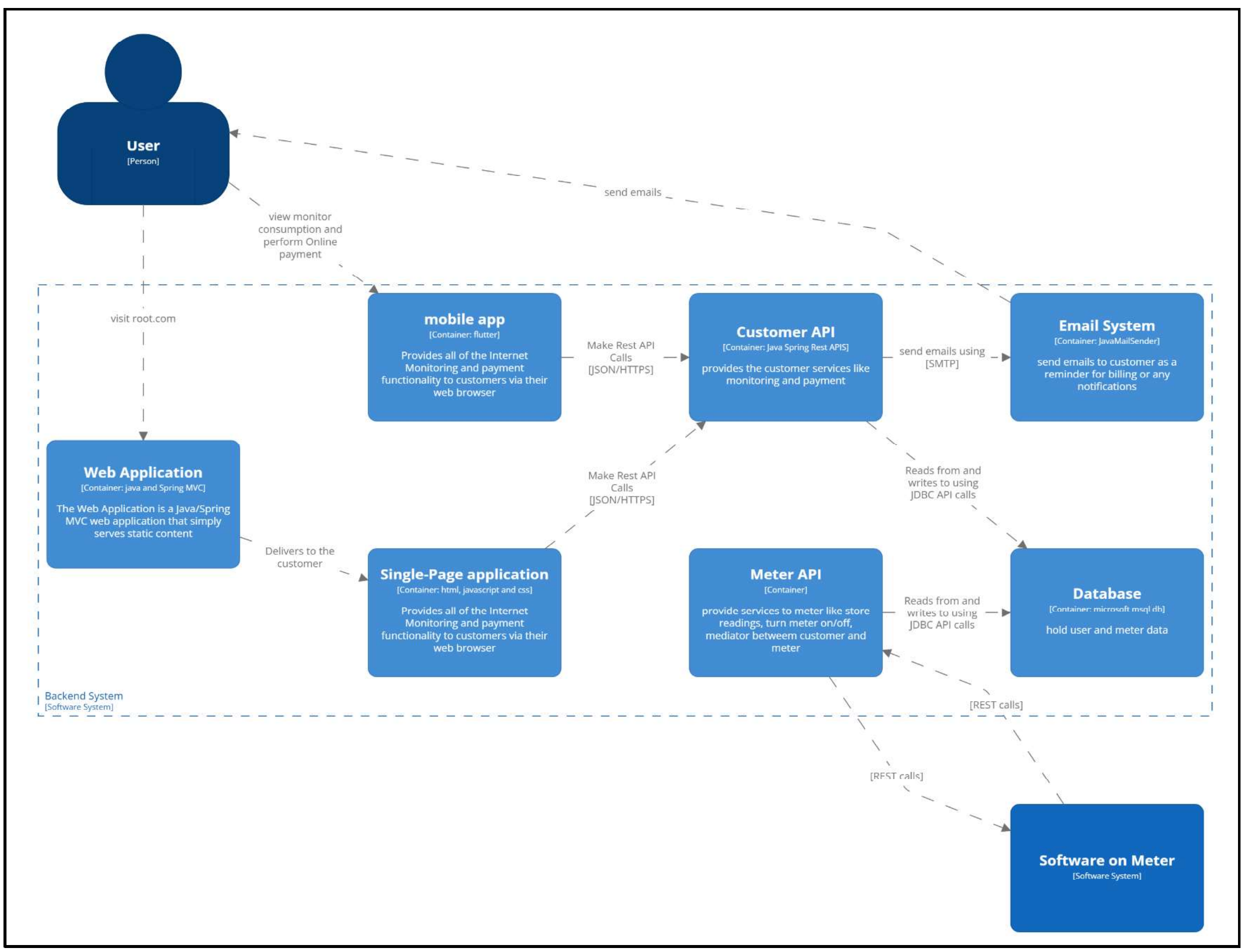


Figure 7.2: Cloud Software System

We use the cloud to provide our services to the clients and to be the mediator between the meter and the client (web app and mobile app).

Our cloud system consists of three major sub-systems:

1- APIs
2- Database
3- Web application

We will discuss each in detail:

- APIs

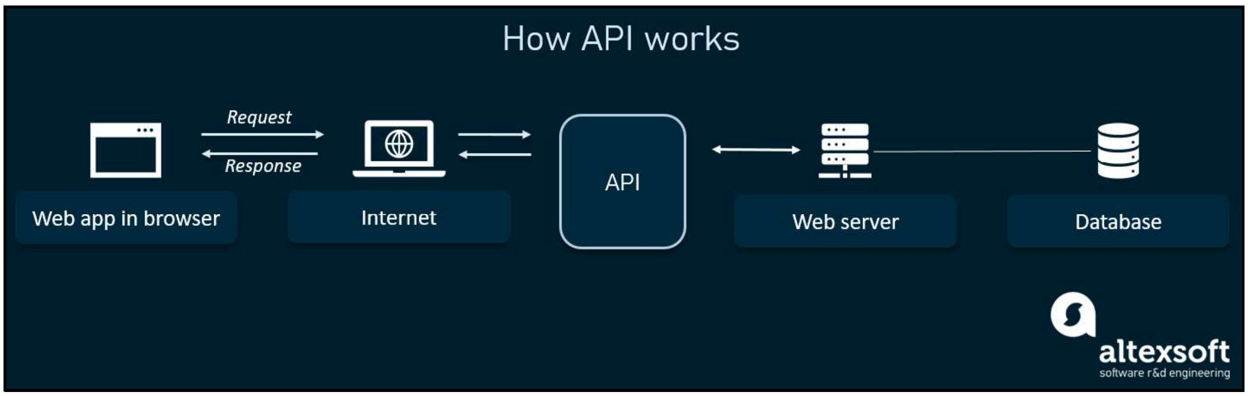


Figure 7.3: APIs Ideas

API or Application Programming Interface. It's the interface between the user who uses the services and the heavy work of our services(server(s) and the system logic).

This contract defines how the two communicate with each other using requests and responses

We build APIs to provide the client(in our case the meter, mobile app, and web app) with the services each one needs.

We‘ve built multiple APIs, for example:

Meter API:
Provides services as:

- Sending meter readings to be saved and performing some financial processing on it to calculate the cost of the consumption of that meter
- Send a Post request to the meter to turn it on or off
- Send a GET request to the meter to get the results of the ML models that run on it.

Consumption API:

- Provides services as:
- Get the consumption of a given user for a specific period of time
- Calculate the meter consumption of money based on the readings sent to it and where the meter is allocated.

User API:

- Provide the user with services like view his information including meter information, consumption information, ..etc .
- Also the capabilities to change any of his saved information on our system, like email, phone number, password, ...etc.

Payment API:

- Perform the payment process by online payment with integration with Stripe(*Stripe is a suite of APIs powering online payment processing*).

Bill API:

- Generate the current bill that contains all the consumption of that user since the last payment date
- Also the capability of providing the user with any past bills he or performed by the date he/she chooses.

Event API:

- Perform the necessary work when tampering happened as our meter is capable of detecting tampering and generating signals that send a request to our system on the cloud, especially the event API to inform the company that there's tampering happened and also provide it with all the required information like which meter, user, and what time stamp of that tampering.

Of course, some of these APIs collaborate with each other to fulfill the request. For example, the payment API uses the billing API to know how much money is in the debt of that user.
Also, the user API uses both the consumption API and meter API, and so on. **[26]**

- Database:

Databases represent the data storage layer for our system on the cloud.

Why do we need a database?

The database is stored for persistence data, and our project has a lot of data that needs to be persisted(saved for a period and not Temporary)

We need a database to:

1- Save the meter's basic information.

2- Save the user's basic information.

3- Save each meter readings.

4- Save the consumption(energy and money) of each meter.

5- Save the payments of users.

The Schema as shown in Fig 7.4:

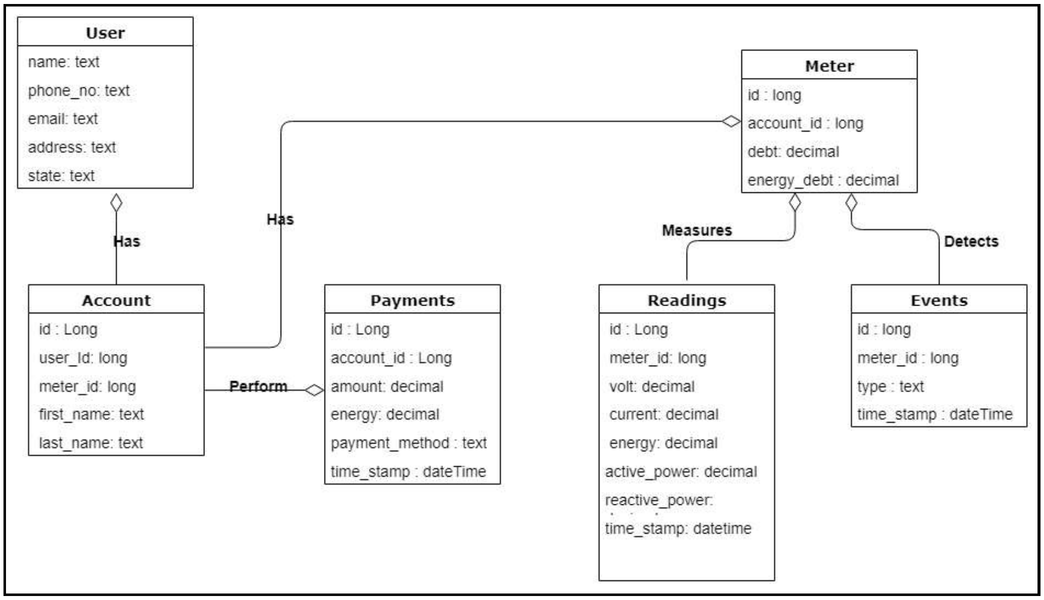


Figure 7.4: Database Scheme

When we started the designing phase of our system we asked ourselves some very important questions:

1- Which database type to use? relational or non-relational database (no SQL)?

2- Do we need to use database replication or not?

3- Do we need database scaling?

Our answers to these questions are based on our needs and our project scope

Our answers to all these three questions:

1- We definitely need a relational database as we deal with relational data like users, meters of these users, readings of these meters, events of these meters, bills of these users, … etc.

2- Database Replication:

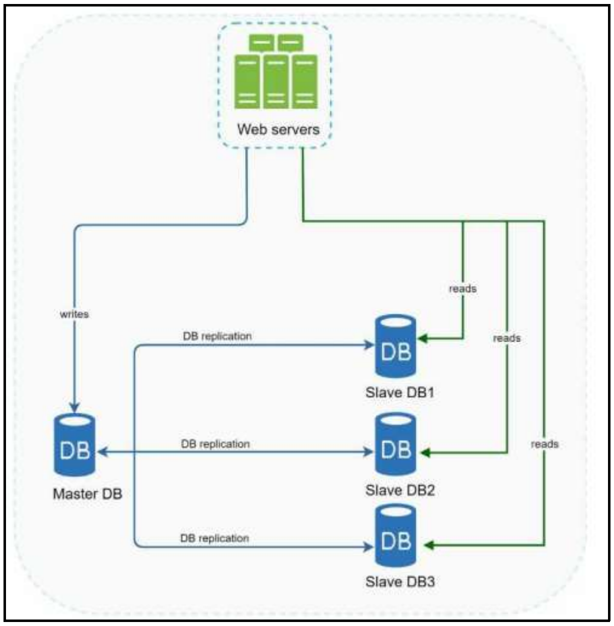


Figure 7.5: Database

In database replication, a master database generally only supports write operations. A slave database gets copies of the data from the master database and only supports read operations. All the data-modifying commands like insert, delete, or update must be sent to the master database. Most applications require a much higher ratio of reads to writes; thus this isn't our case as we write more than we read from the database. **[16]**

So we don't need database replication.

3- Database scaling

There Are two types of scaling:

- Vertical scaling
- Horizontal scaling

Vertical scaling:

Vertical scaling is about adding more hardware (CPU, rams,.. etc.)

But it has limitations as at some point when the users and requests become very large(millions) and you need to add more hardware and of course, there are hardware limitations.

Drawbacks:

- Greater risk of a single point of failure.
- The overall cost of vertical scaling is high. Powerful servers are much more expensive.

Horizontal scaling:

Also known as sharding, it is the process of adding more servers.

Fig 7.6 shows the difference between vertical and horizontal scaling.

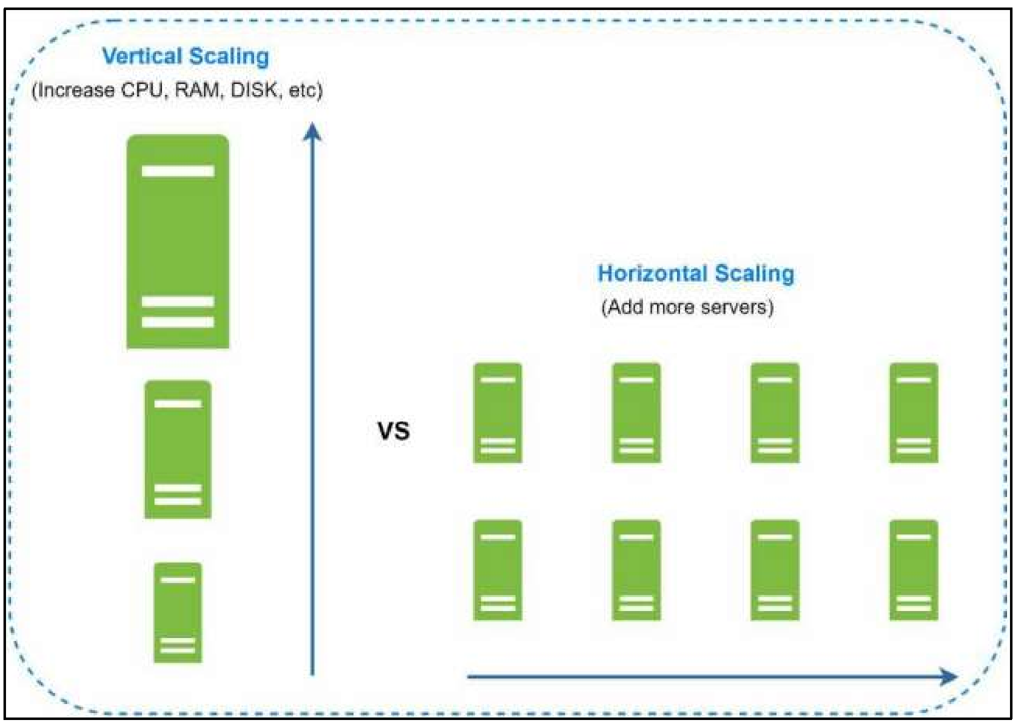


Figure 7.6: Vertical and Horizontal Scaling

Because we have millions of users and a very large number of requests to the database, we will need to share our database as shown in **Fig 7.7**.

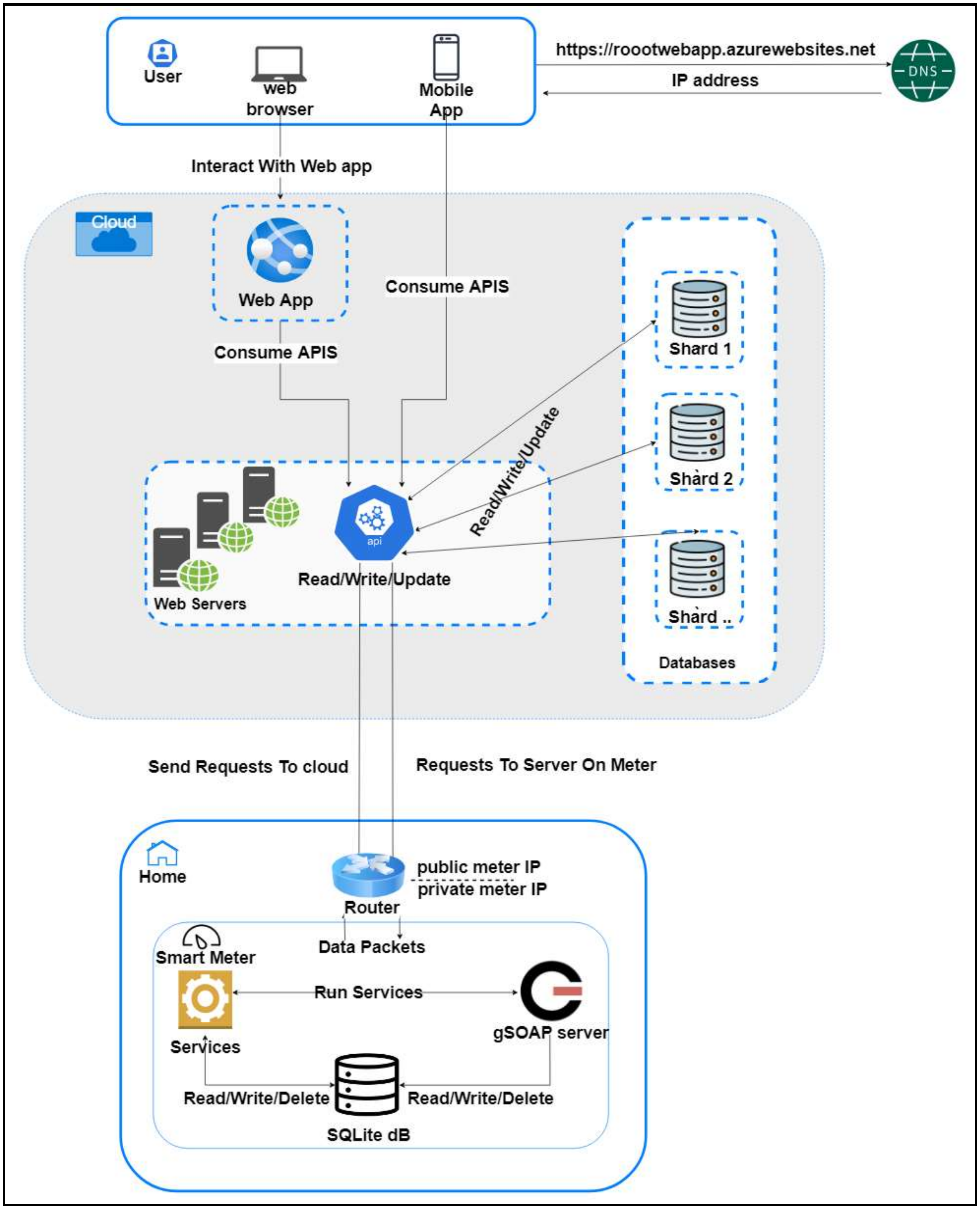


Figure 7.7: Sharding our Database

Fig 7.8 shows an example of sharded databases. User data is allocated to a database server based on user IDs. Anytime you access data, a hash function is used to find the corresponding shard. In our example, user_id % 4 is used as the hash function. If the result equals to 0, shard 0 is used to store and fetch data. If the result equals 1, shard 1 is used.

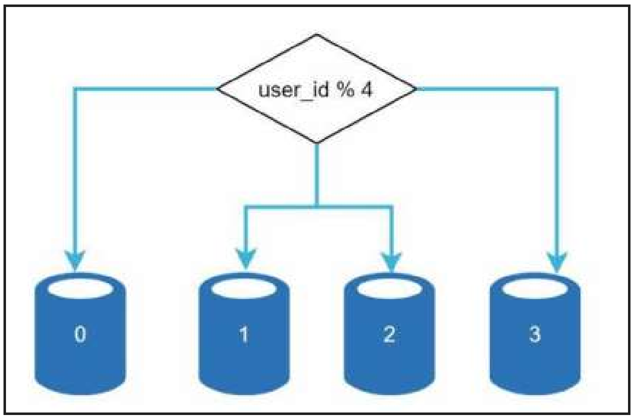


Figure 7.8: Sharded Databases

- Web Application

Overview:

Our web app consumes the APIs we‘ve built to provide the user with all the functionality and services that the user might think of or need to guarantee the highest level of user satisfaction.

Design:

Our web app is based on spring MVC.
We‘ve used thymeleaf to render the HTML pages(templates) from the server and binding data models to it.

We have used the MVC which is a design pattern that emphasizes the separation between the software's business logic and display. This "separation of concerns'' provides for a better division of labor and improved maintenance.

MVC stands for Model, View, Controller.

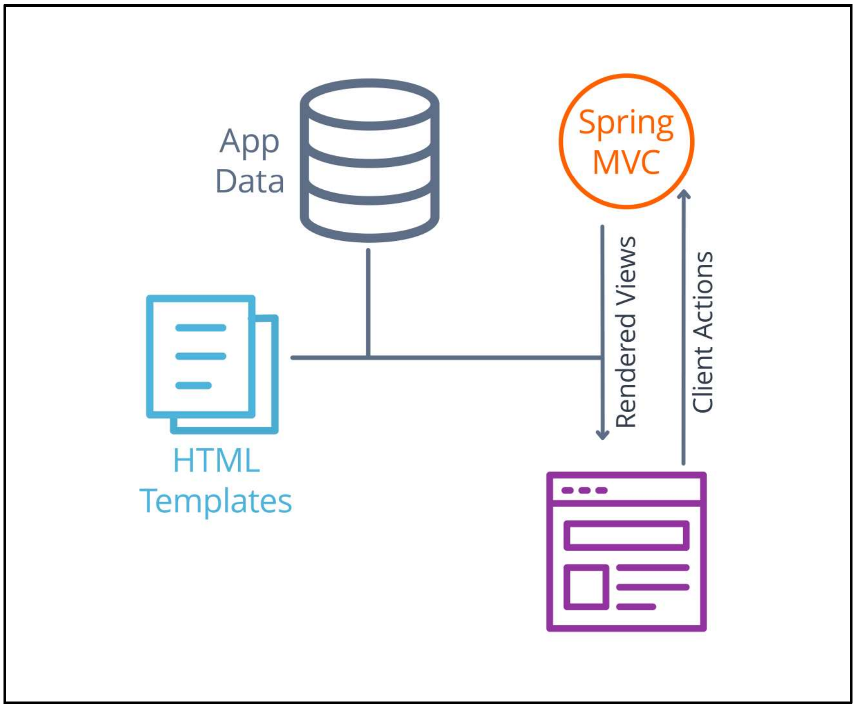


Figure 7.9: MVC

The Relationship Between Spring MVC and the Client:

The figure above shows the relationship between Spring MVC and the client. Spring MVC uses HTML templates and application data to render a view for the client browser to display. The view exposes certain actions to the user, which when triggered are sent to Spring MVC, which processes the actions and renders new views for the client.

Interacting with a simple web application:

In a web application, there are two components: the client that sends HTTP requests, and the server, which sends HTTP responses back. In the case of a web browser client, the responses the server sends need to be in the format of HTML, the document language of the web. The HTML that is sent to the client both defines the data that the user sees, as well as the actions a user can take - things like buttons, links, and input forms are all part of what the server is responsible for generating.

HTML Templates:

HTML templates are essentially just HTML files with special tags and attributes that can be combined with a data model by a template engine like Thymeleaf to generate fully functional HTML documents. Spring MVC provides a standard way to choose a template and supply the necessary data model when a request comes in, which allows for a truly dynamic user experience.

The data model defines how the application stores and retrieves (serializes and deserializes) the Java objects into/from the database. Data modeling starts with designing the database tables which are analogous to the Java classes ("model" classes of the MVC).

7.3.2 Meter Software System

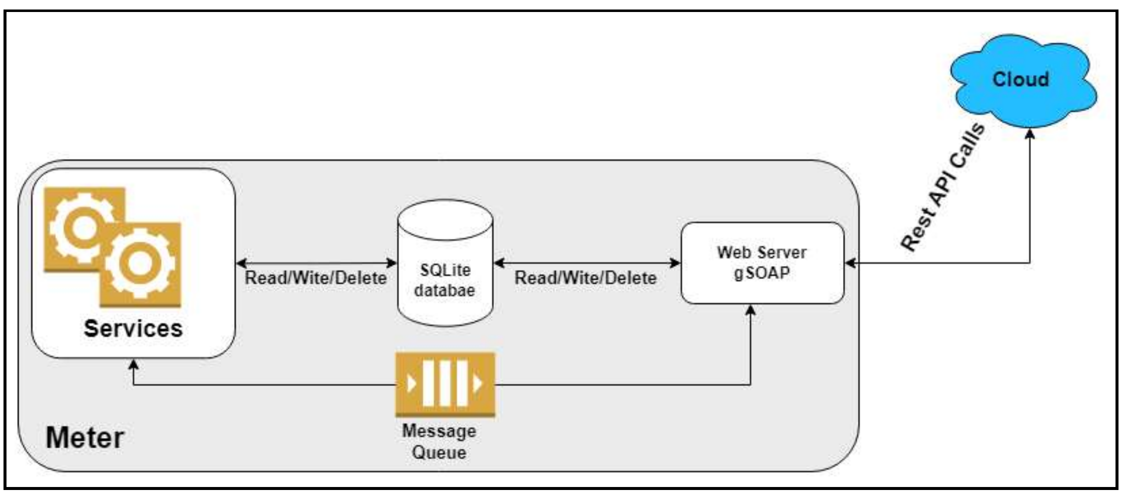


Figure 7.10: NGSM Software

In **Fig 7.10** we can see we have 4 main components:

1- Web server

2- SQLite DataBase

3- Services

4- Message Queue

- ## Web Server and gSOAP:

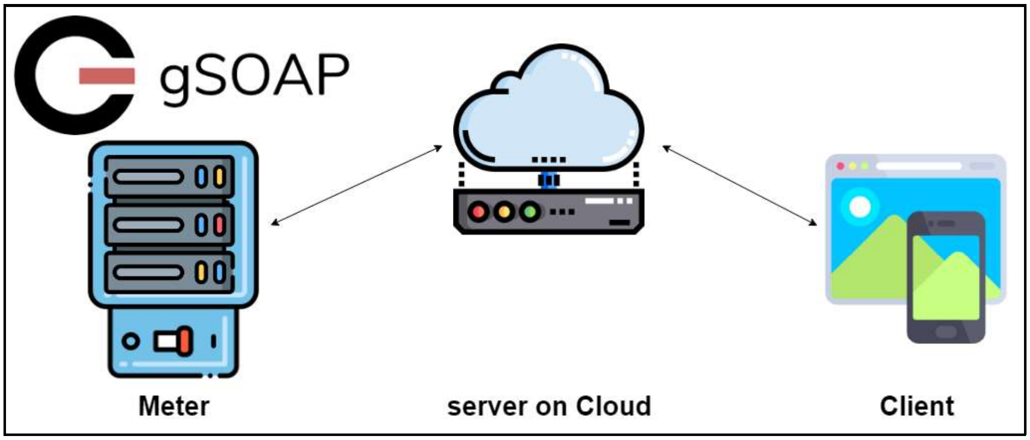


Figure 7.11: Server Communication

As shown in **Fig 7.11**, we need to be able to communicate with the meter and the meter to be able to communicate with us.

So we had the need to implement a web server on the meter itself.

As usual, we needed to ask ourselves in the designing phase some questions:

Q1- Which web server to use?

Q2-which programming language to use?

Q3- Do we need a toolkit or framework?

Q4- which database to use?

Our answers to these questions are based on our needs and our project scope

A1- our project is embedded Linux which is targeted to run on the f1c200, so we need the possible lightest web server which is lighttpd.

A2- Because we have very limited resources, we need the lightest possible programming language which is C.

A3- yes we need a toolkit as we use C as a programming language some low-level layers will be too hard and time-consuming to handle, so we will use gSOAP as the toolkit for C.

A4- we need to save relational data so we will use a relational database NOT NoSQL.

Also, we need the lightest possible database to run locally on the meter, so our choice is SQLite.

- ### Database on meter:

We need a local database to save readings and events temporarily until sending them to the server on the cloud.

Readings	
ROWID	Long
Volt	Double
Current	Double
Energy	Double
Power	Double
Time_Stamp	Text

Events	
ROWID	Long
Event_type	Text
Time_stamp	Text

Figure 7.12: Database Tables

As we mention in the 7.3.1 section, we used SQLite database because it's a very lite weighted and relational database.

- ### Services:

We used the daemons' architecture.

Daemon is a computer program that runs as a background process, rather than being under the direct control of an interactive user. When the system boot is complete, the system initialization process starts spawning (creating) daemons through a method called forking, eliminating the need for a terminal (this is what is meant by no controlling terminal).

We choose this architecture for many reasons:

1- Our services work in the background.
2- We have more than one service.
3- Our services are always running.
4- If one service faced an error, it could restart itself.

- ### Message Queue:

A message queue is a durable component, stored in memory, that supports asynchronous communication.
It serves as a buffer and distributes asynchronous requests.

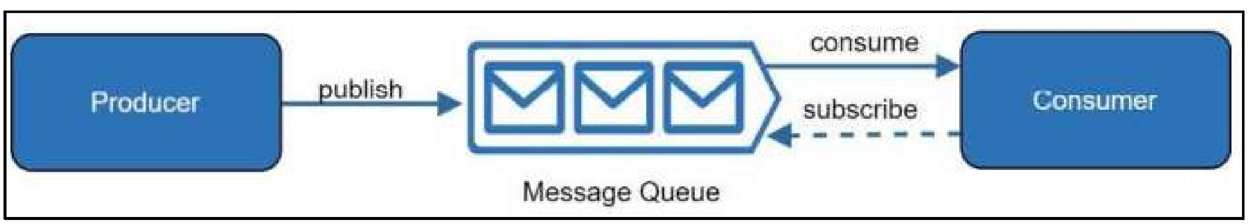


Figure 7.13: Message Queue

With the message queue, a producer can send a message to the consumer when the consumer isn't available, and vice-versa.

We have faced a problem that required us to use the message queue architecture.
The problem is DATABASE LOCKING!!
When we try to perform concurrency operations on the database, the database is locked until one finishes, then unlocked to let the other operation operate.

That happened when we tried to read and write from and to the same database table “Readings” at the same time. As shown in **Fig 7.14**.

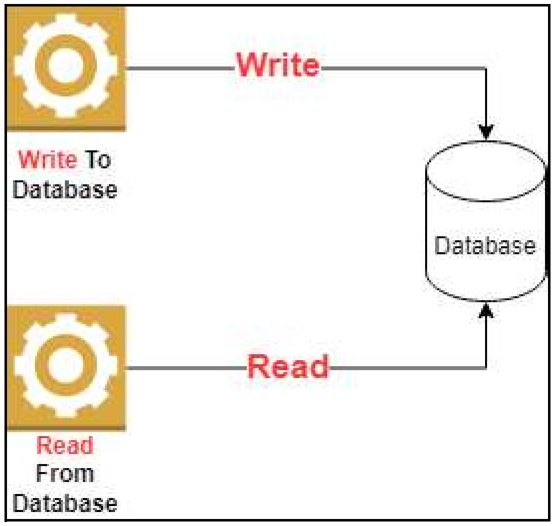


Figure 7.14: Database

So, the solution is to use a message queue to hold the new reading values that are supposed to be saved to the database but are locked because another service (read from the database) is running,
And send to be sent to the server directly as shown in **Fig 7.15**.

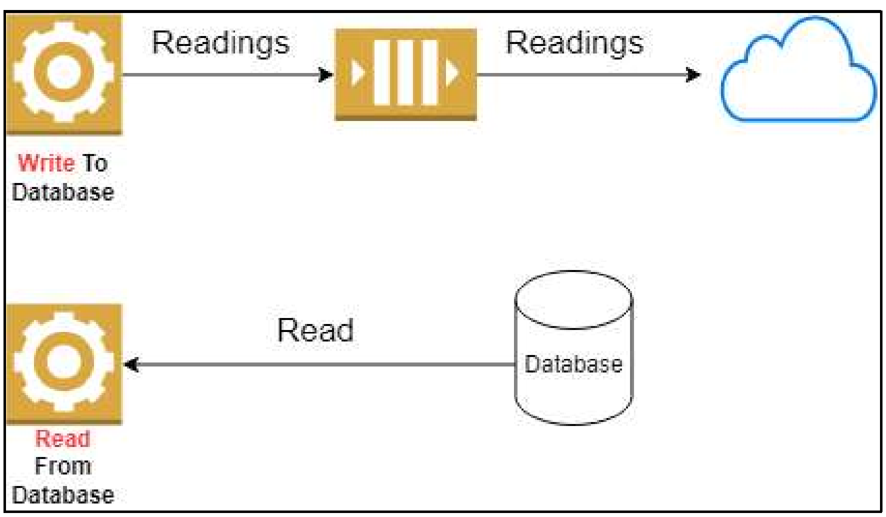


Figure 7.15: Database

We ‘ve used another more affecting solution that fixes our database locking problem.

As mentioned in SQLite documentation here: https://sqlite.org/wal.html

“The default method by which SQLite implements atomic commit and rollback is a rollback journal. Beginning with version 3.7.0 (2010-07-21), a new "Write-Ahead Log" option (hereafter referred to as "WAL") is available.

There are advantages and disadvantages to using WAL instead of a rollback journal. Advantages include:

1. WAL is significantly faster in most scenarios.

2. WAL provides more concurrency as readers do not block writers and a writer does not block readers. Reading and writing can proceed concurrently.

3. Disk I/O operations tend to be more sequential using WAL.

4. WAL uses many fewer fsync() operations and is thus less vulnerable to problems on systems where the fsync() system call is broken. ”

Beginning with version 3.7.0, a new Write Ahead Logging (WAL) option is available, in which reading and writing can proceed concurrently.

By default, WAL is not enabled. To turn WAL on, we need to configure it like:

```
//enable WAL to support concurrency

sqlite_3_exec(db, "sqlite3_exec(db, "PRAGMA journal_mode =
WAL;", NULL, 0, 0);
```

7.4 Security

Security on Transient:

Secure communication between meter, head-end system, and the client(mobile/browser), we deal with sensitive information, like credit card no, passwords, and sensitive commands like turning the meter ON/OFF.

So, we need to ensure:

- Confidentiality: ensure data is encrypted
- Integrity: ensure data is correct
- availability

By making the communication over HTTPS and using digital certificates.

We use HTTPS to ensure every data is encrypted as shown in **Fig 7.16**.

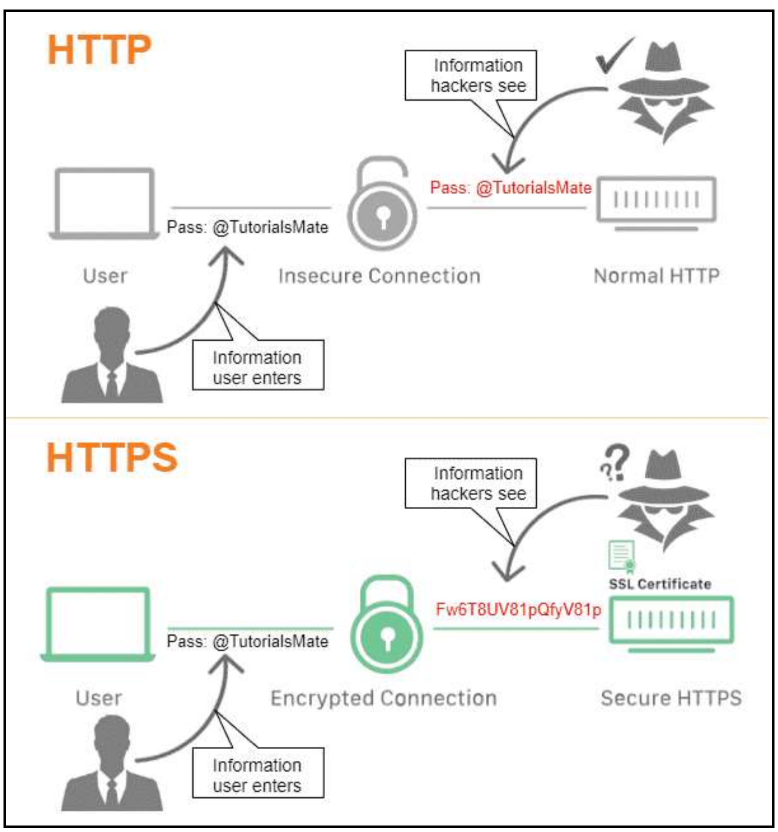


Figure 7.16: HTTPS Encryption

Digital Certificates:

We use digital certificates to prevent man-in-the-middle attacks by ensuring the public key that is sent back to the server actually belongs to the intended server. As shown in **Fig 7.17**.

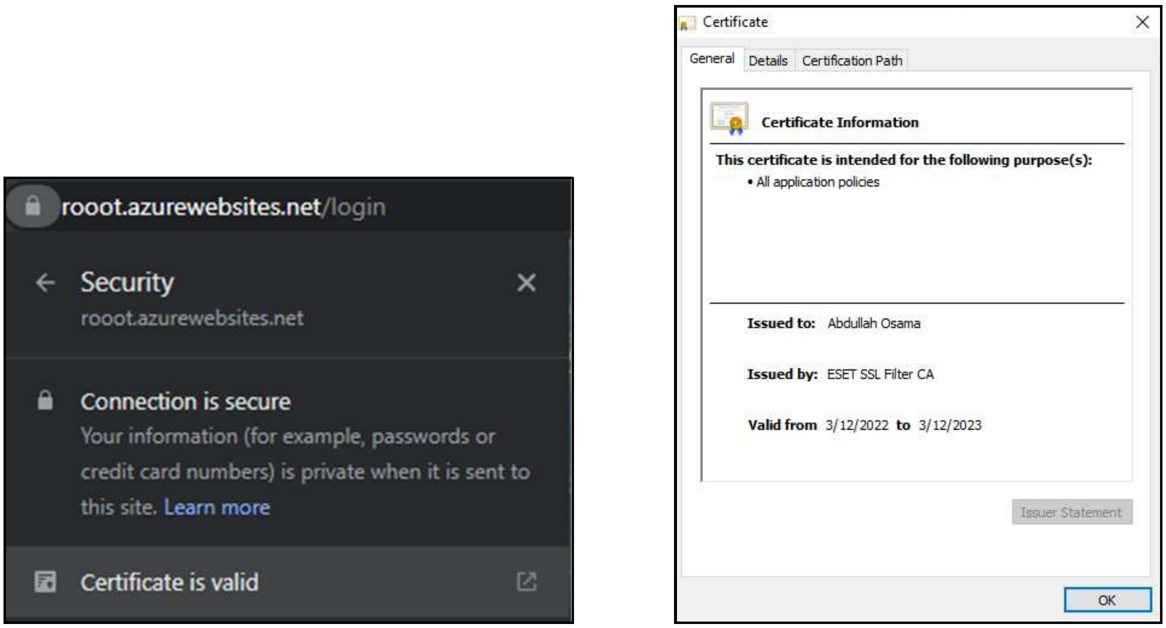


Figure 7.17: Certificate

We have used two certificates, one on the back-end(head-end system) and the other one on the meter to prove its identity.

Data Exposing:

Sometimes when servers respond with object JSON data to the outside world(front end(HTML templates or mobile app) which can take formate like:

```
{
    Id : id,
    Meter_id : meterid,
    Name:name,
    Password:password,
    Address:address
}
```

This exposes sensitive information like the user's password.
So we need a way to return only the necessary information, not less and not more.

Data Transfer Objects (DTOs)

Data structures designed to represent the needs of the front end.

What DTOs give to us:

- Simplify and document interaction between front end and Controller.
- Conceal database structures.
- Limit the amount of data exchanged.
- Customize display data to meet the needs of the front end without exposing the sensitive data.

Example of a DTO on the same user object will be like:

```
{
	Id : id,
	Meter_id : meterid,
	Name:name,
	Address:address
}
```

We can see now the password field no longer exists.

So, by taking advantage of the MVC architecture that we used, we can make the conversion between the normal entity and DTO in the controller layer to return to the representation layer only what it needs. As shown in **Fig 7.18**

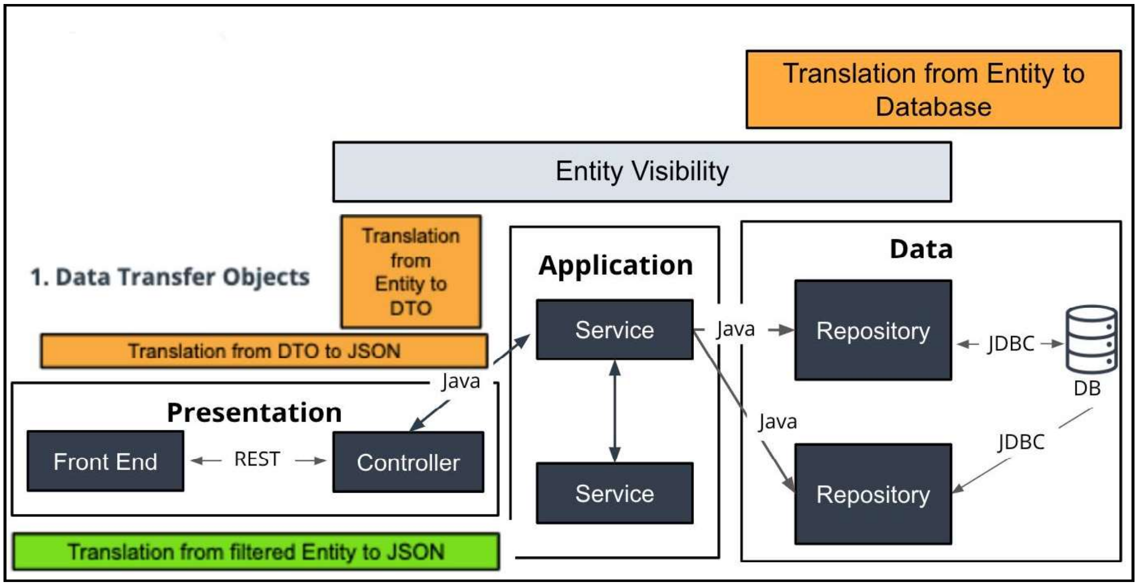


Figure 7.18: Security

Security on Rest:

We've used the Bcrypt hashing algorithm to encrypt users' passwords to store them in the database.

Also used Microsoft azure SQL database that ensures that every piece of information in the database is encrypted.

Security of the REST APIs:

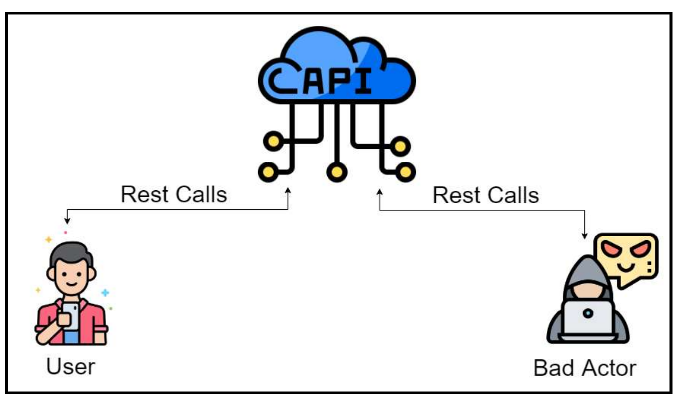


Figure 7.19: Rest APIs

To authenticate the REST requests to our APIs we need to restrict access to the APIs to only the authorized ones. We faced a limitation due to using HTTPS because it is a stateless protocol, so we need to find a way to authenticate each request without causing unneeded latency.

Basic Auth:

On the way -which is the worst one- is to use basic auth, which is to send the username and password with each request, and the server checks if this credential is right or not.

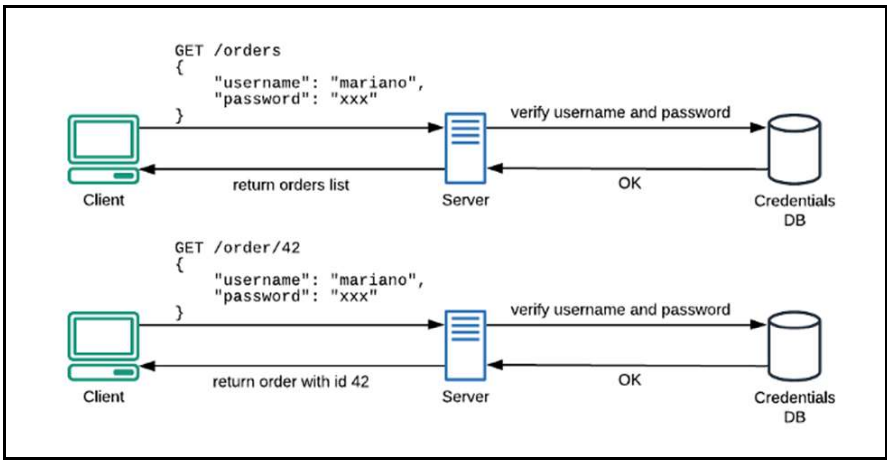


Figure 7.20: Basic Authentication

Problems:

- Send a password with each request which increases the vulnerability to attacks.
- Hitting the database with each request to check in the database the username and password which was sent in each request.

Server-Side Session:

An alternative solution is using a server-side session:

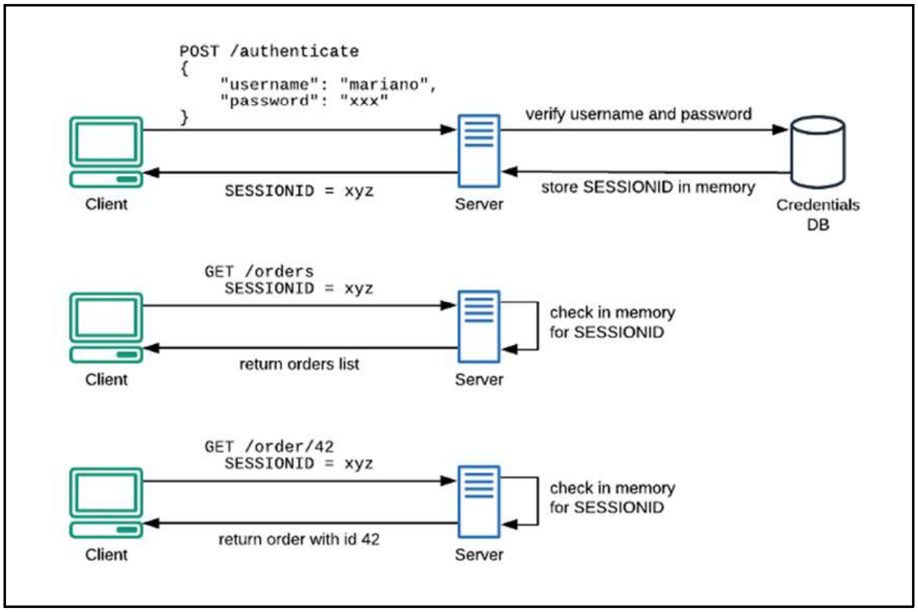


Figure 7.21: Old way using server-side session

It is better than the basic auth but still has the hitting database problem with each request.

Tokens:

The best solution for our case is by using **JWT** (JSON WEB TOKEN). It defines a compact and self-contained way for securely transmitting information between parties as a JSON object.

Because of its relatively small size, a JWT can be sent through a URL, through a POST parameter, or inside an HTTP header, and it is transmitted quickly. A JWT contains all the required information about an entity to avoid querying a database more than once. The recipient of a JWT also does not need to call a server to validate the token.

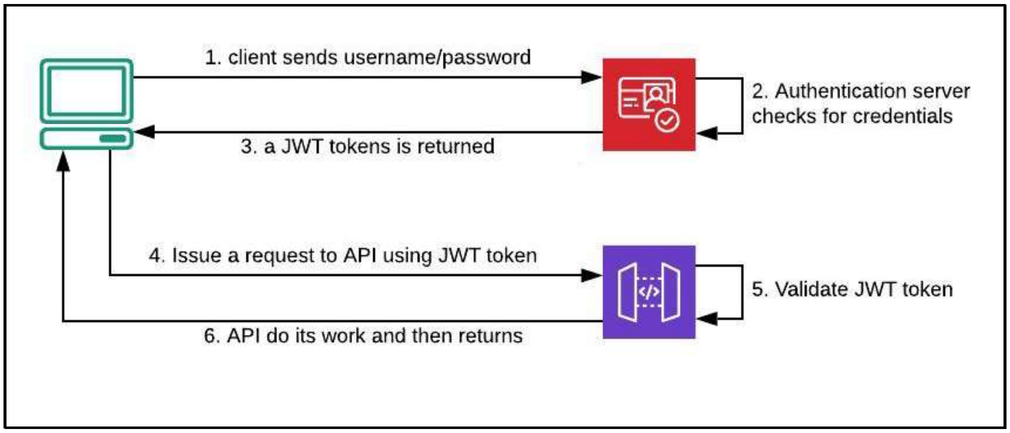


Figure 7.22: The Solution is JWT

Pros:

- No need to hit the database to verify the JWT.
- Ensure integrity as it has a signature.
- JWT is *self-contained*, which means it contains every information needed to allow or deny any given requests to an API.
- JWT is stateless by design, so we don't have to fight with the stateless design of HTTP.
- Authentication: When a user successfully logs in using their credentials, an ID token is returned. According to the OpenID Connect (OIDC) specs, an ID token is always a JWT.
- Authorization: Once a user is successfully logged in, an application may request to access routes, services, or resources (e.g., APIs) on behalf of that user. To do so, every request must pass an Access Token, which may be in the form of a JWT. Single Sign-on (SSO) widely uses JWT because of the small overhead of the format, and its ability to easily be used across different domains.
- Information Exchange: JWTs are a good way of securely transmitting information between parties because they can be signed, which means you can be sure that the senders are who they say they are. Additionally,

the structure of a JWT allows you to verify that the content hasn't been tampered with.

Therefore, our tokens will look like this:

Header.payload.signature

Header: contains metadata about the type of token and the cryptographic algorithms used to secure its contents.

Payload (set of claims): contains verifiable security statements, such as the identity of the user and the permissions they are allowed.

Signature: used to validate that the token is trustworthy and has not been tampered with. When you use a JWT, you must check its signature before storing and using it.

Pseudo code illustrates the Operation of JWT

```
//Pseudo code of trying to authenticate request

Function authenticate (name){

START:

User = load from db user by username (name)

IF user is exist :

    IF Compare entered pass with the stored is TRUE :

        generate_JWT()

    ELSE:

        print "Authentication Failed"
```

```
Else:
        Print "user not exist"
END IF
END
//pseudo code to generate jwt
Function generate JWT(username){
START:
Create subject(username)
Set expiration date
Sign with HMAC512
END
}
//pseudo code illustrates server behavior when receiving a request
Function server_request(request){
BEGIN
Jwt ← extract JWT from Request Header
IF jwt is valid:
        Serve user request()
ELSE:
        PRINT "UNAUTHORIZED"
END IF
```

END

}

JWT Validation:

We don't validate the JWT by ourselves as it discourages doing manual JWT validation since it might be easy to improperly implement and miss some important details that will lead to serious security vulnerabilities. Most JWT libraries take care of JWT validation for us.

We've used the DefaultJwtSignatureValidator class from the dependency io.jsonwebtoken library to validate the JWT.

Snapshot of our API using JWT(Access Token):

1- First log in correctly and get the JWT in the success response.

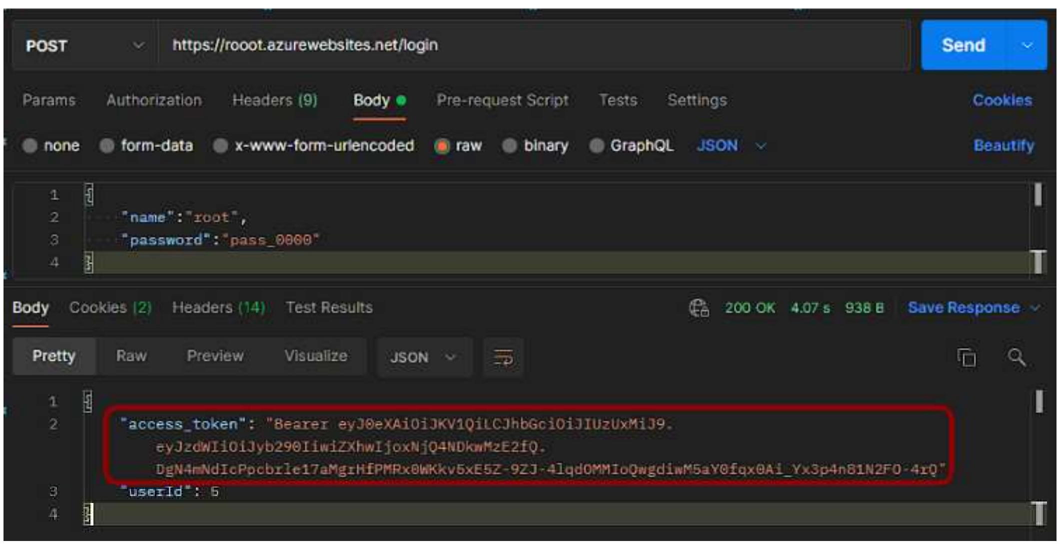


Figure 7.23: Success Login and get the JWT

2- Once we get the JWT from the server we can send it with each request:

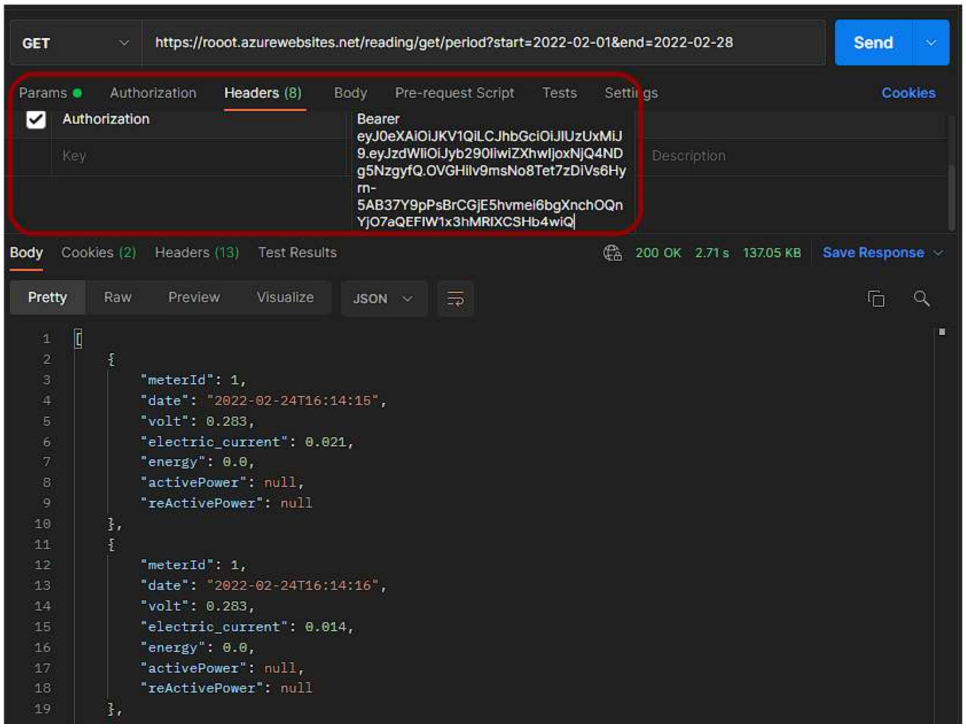


Figure 7.24: Send JWT with Each request

7.5 Tariff and User Profiles

Today's interconnected power systems supply a number of consumers. With such a big organization, management, economy and control come into account automatically. The supply companies (usually in the public sector) have to sell their electricity at such a rate that it covers the costs of generation, transmission, distribution, the salaries of the employees, the interest and depreciation and the profit targeted by the company. This rate at which electrical energy is sold to the consumers is termed as 'tariff.'

The cost of generation of electricity will depend upon various factors such as Connected Load, Maximum Demand, Load factor, Demand Factor, Diversity Factor, Plant Capacity Factor and Use Factor (learn more about these factors). These, in turn, will depend upon the type of load and load conditions. Hence, the tariff is different for different types of loads (and hence different consumers).

Therefore, while fixing the tariff, we have to consider various consumers (industrial, domestic, commercial, etc.) and their requirements. Due to this, the whole process becomes complicated.

Various Types Of Electricity Tariff

1. Simple Tariff

In this type of tariff, a fixed rate is applied for each unit of the energy consumed. It is also known as a uniform tariff. The **rate per unit of energy does not depend upon the quantity** of energy used by a consumer. The price per unit (1 kWh) of energy is constant. This energy consumed by the consumer is recorded by the energy meters.
Graphically, it can be represented as follows:

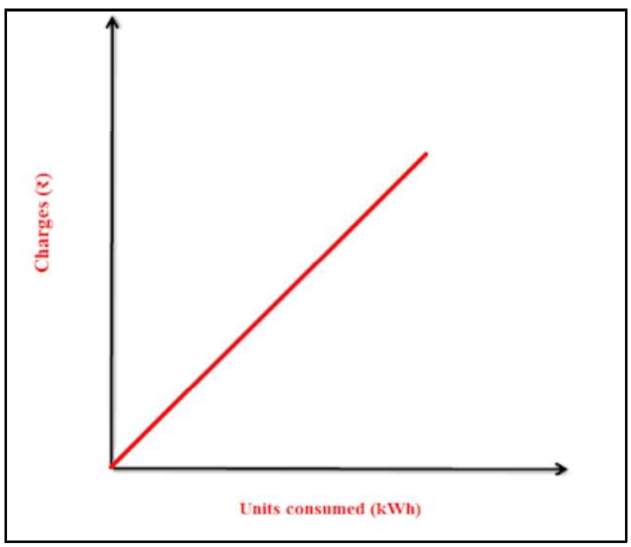


Figure 7.25: Simple tariff

2. Flat Rate Tariff

In this tariff, different types of consumers are charged at different rates of cost per unit (1kWh) of electrical energy consumed. Different consumers are grouped under different categories. Then, each category is charged money at a fixed rate similar to Simple Tariff. The different rates are decided according to the consumers, their loads and load factors.
Graphically, it can be represented as follows:

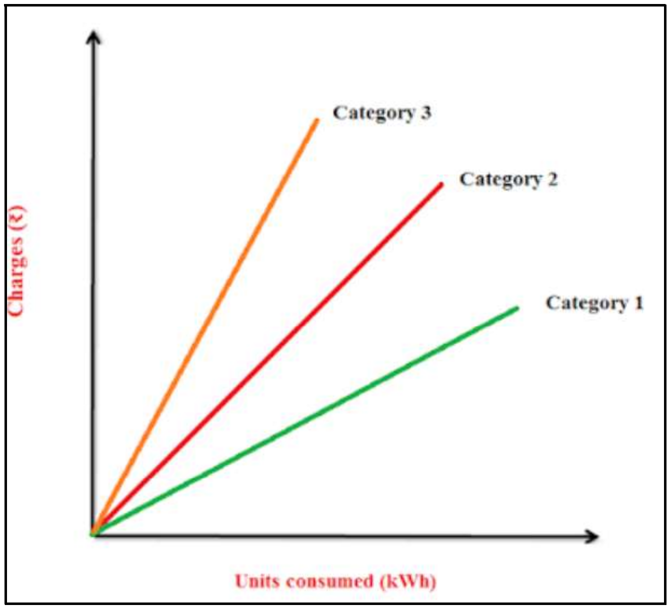


Figure 7.26: Flat rate tariff

8. NGSM Web-App System

8.1 Overview and Summary

The website is a simple way to show our work in this project. This site would help the user to communicate easily with the utility. The user could visit the site at the landing page and then by using his specific username and password can access his account and interact easily with his data in detail.

This website provides the user with:

- Viewing the **'Home'** page that helps the user to understand more about the project and the benefits of the smart meters.
- Viewing the **'About Us'** page to know more about our team.
- Viewing the **'Frequently Asked Questions (FAQs)'** to know more about the system work.
- Viewing the **'Pricing'** page.
- Logging in by the specific username and password from the **'Login'** page
- Showing his information.
- Showing the energy consumption in the form of a graph in the **'Chart'** page during a specific period that the user could determine.
- Pay the bill online from the **'Payment'** page.
- Logout from your account.

8.2 Web-App System Block Diagram

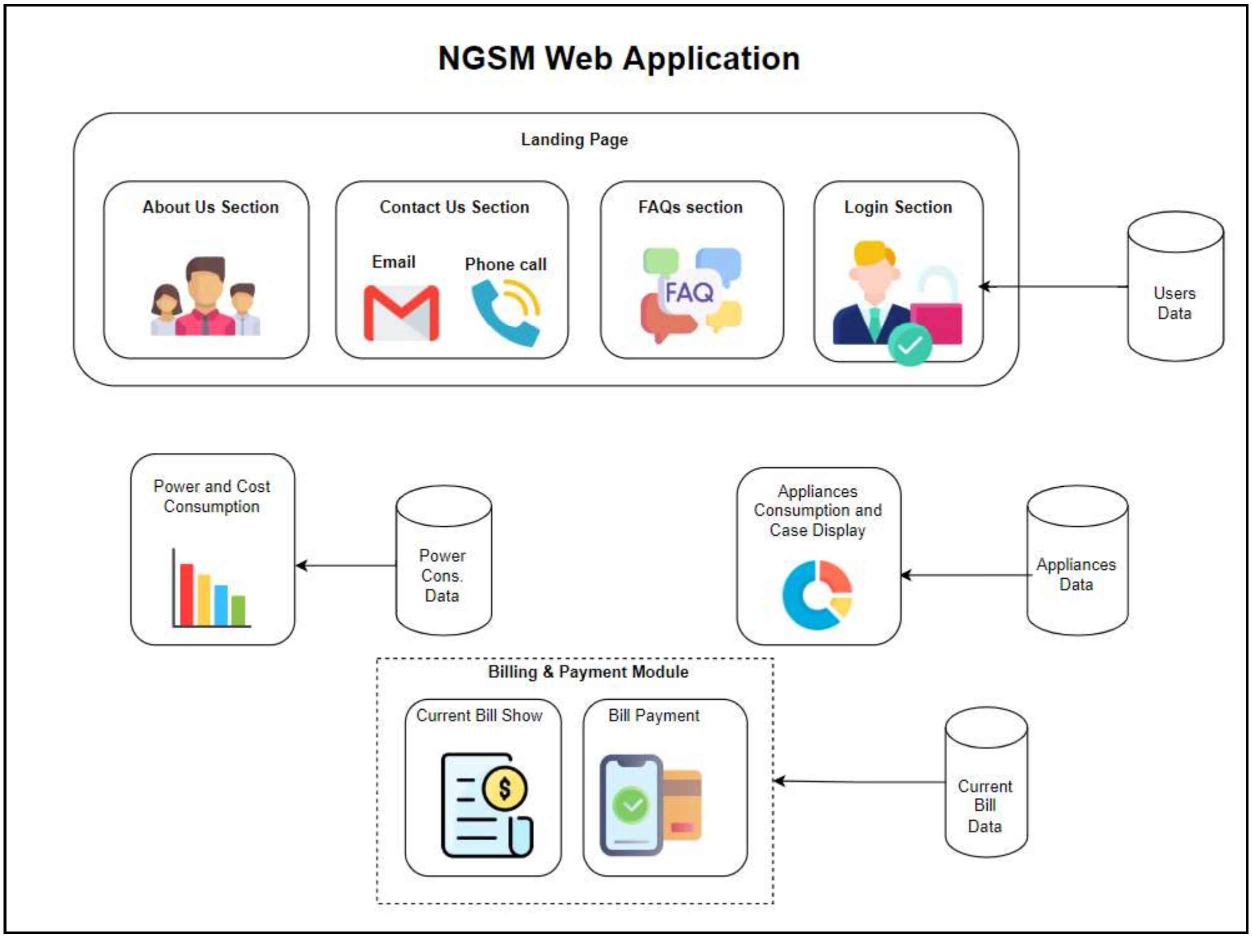


Figure 8.1: Web-Based System block diagram.

8.3 Main Components and Features

The main components of the website:

- Web-Hosting

Every website contains numerous files and envelopes in the backend that enable the site to be accessed by anyone. Additionally, those documents ought to be stored in a location with web-enabled. You need a dependable hosting service because hosting affects how well a site performs from many angles.

- **Layout**

The layout or structure of the website refers to the website's hierarchy. This is where the page is decided to be the landing page and which pages you can directly access from there. Another aspect of the layout works with a wireframe or an internal map of one specific web page. Everything that appears on a web page is dictated and placed through a wireframe. For example, let's say your webpage has a header, text, a corresponding picture, a notification that your website uses cookies, and an ad at the bottom. These pieces are called elements, and a wireframe holds the elements in place during the design.

- **Content**

Content is what your users are there to see. Text, pictures, videos, downloadable files, these all count as your content. Content is deeply intertwined with your design since the design is what makes the content appealing, but at the end, you'll need to provide the users with rich, informative content that gives them the answers they need.

- **Optimized web pages**

One of the most frustrating experiences visitors face is waiting for a page to load. With people accessing on various platforms, it’s only natural that they expect a fast result for the content they’re viewing. One way to improve the page speed is to compress all your images before loading them to your website. Large image files are one of the leading causes of slow pages. **[27]**

The website features:

1. Landing page:

The landing page is a dedicated page on a website that you land on directly to help the user to know more information about the system, the benefits of using smart meters, and the features provided by the smart meter.

2. About page:

This page allows you to showcase the business’s strengths, provide essential information about our team, and our work, and generally summarize the unique identity.

3. Frequently asked question page:

Frequently asked questions, or FAQs as they are known, are a great way to improve the customer's experience. It allows the user to view the answers to the questions that are most commonly asked about our service.

4. Pricing page:

This page helps the users to expect the average consumption, this will allow the users to predict their upcoming bills.

5. Login page:

This secured page enables only users with the correct username and password to log in and interact with their data.

6. View your information:

After logging in, your information will be viewed to check it.

7. View consumption page:

This page enables the user to choose a starting date and ending date to view the energy consumption and the amount of money paid during this period. This will be useful for expecting the upcoming consumption and in order to help the consumer to reduce his consumption.

8. Devices consumption page:

This page enables the user to see his consumption during a specific period on an appliance. This feature helps the consumer to know which devices consumed more energy to reduce using them.

9. Payment.

This page shows the user his bill to enable him to pay online by his credit card. After entering his credit card's information, the user could pay his bill. After finishing the payment operation, a page will be shown to the user by a message **'Success'** or **'Failed'** operation. And then the user can load any page of the previous pages in the website.

8.4 Grid Communication

Using APIs in the web application to perform the desired actions. We need APIs in many features.

- **Login**, when the user enters his username and password and clicks 'sign in' a request will be sent to this API, containing the entered username and password to make sure that they are correct, if so, the user will get a response with the user id and login successfully.
- **Show energy and power consumptions**, the user can select different start & end dates through the date pickers, these dates are put in this API, so, after pressing the button "Show Consumption" or "Show Cost", the request will be sent to the previous API with the selected dates.
- **Show the user's bill**, As a request will be sent to this API containing the user's id, then, the user will receive a response containing his bill.
- **Show devices' consumptions**, An API is used to show the power consumption of each appliance, by sending a request to this API, the user will get a response containing each appliance name, consumed power and case to show if it's ON or OFF.

8.5 GUI Design and Procedure of Operation

This section illustrates the frontend design that the user will use and how to use it. Firstly when visiting our website the landing page will be opened. The design and the features in this page shown in the following paragraph.

- Landing page

Once the user visits our site, the landing page shown in **Fig 8.2** will appear containing a navigation bar (nav-bar). This nav-bar contains ,at the left corner, our logo and the project name "Next Generation Smart Grid Meter (NGSM)". In the right corner, there are several tabs that navigate the user to other pages such as "About Us", "FAQs", "Pricing" and "Login" page.

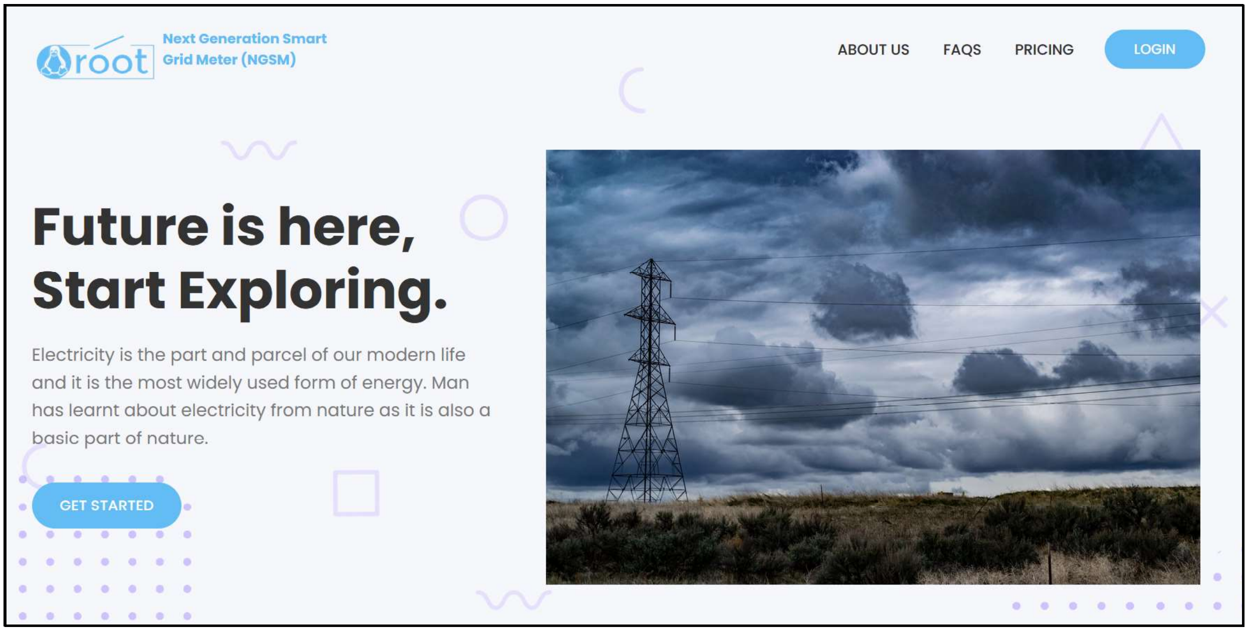


Figure 8.2: Landing page.

After scrolling to the next section, you will find an overview section that illustrates our idea of this project **"Next Generation Smart Grid Meter (NGSM)"** in **Fig 8.3.**

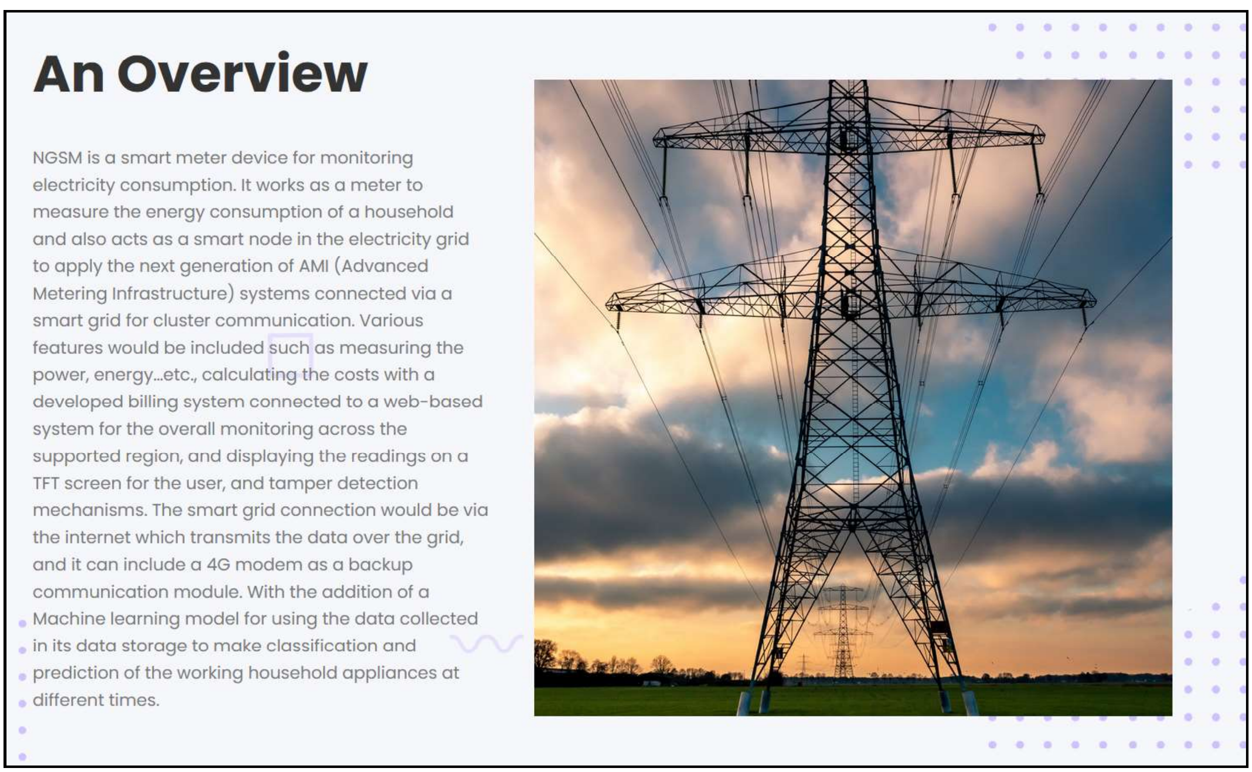


Figure 8.3: Overview section

When scrolling down, you will find the features of our system as shown in **Fig 8.4.** The system includes Mobile application, web-based platform and Machine Learning. All these features aim to help the user for ease of use of the energy and to reduce his consumption.

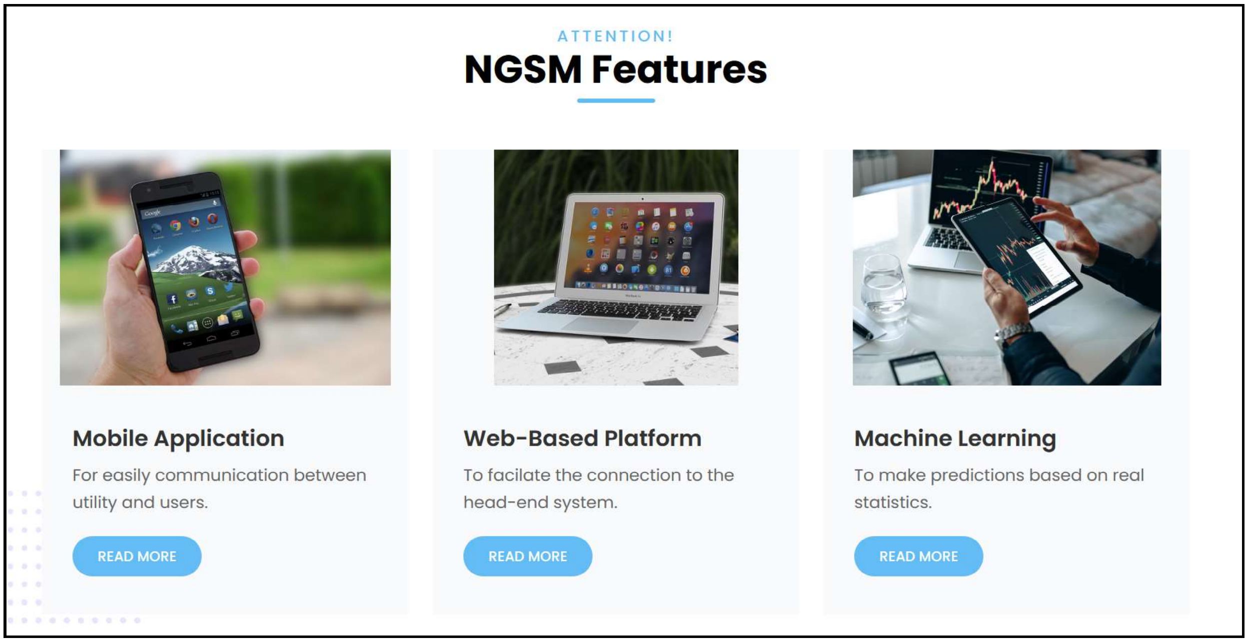


Figure 8.4: Features of the smart meter

When you scroll down, the services that the smart meters offer to the users and the utility will appear as shown in **Fig 8.5**.

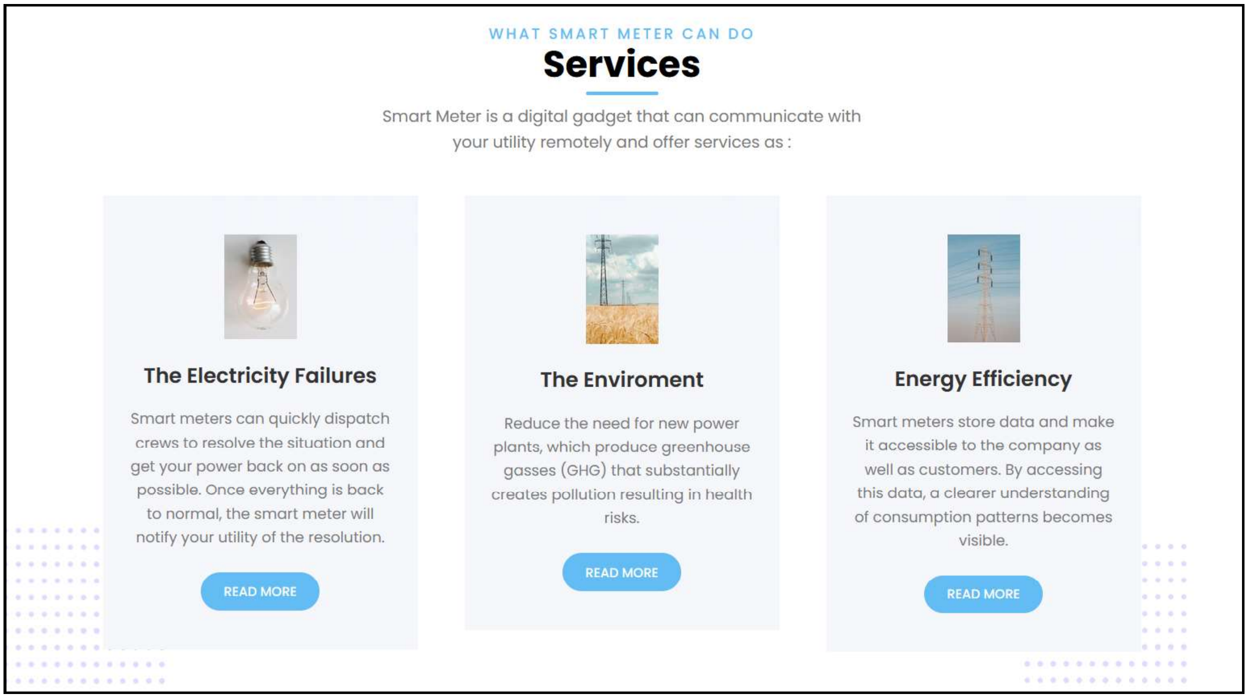


Figure 8.5: Smart meter services.

When you continue scrolling, you will find a section that enables the user to contact the utility even by a message or by a mobile call. This Contact Us section is shown in **Fig 8.6**.

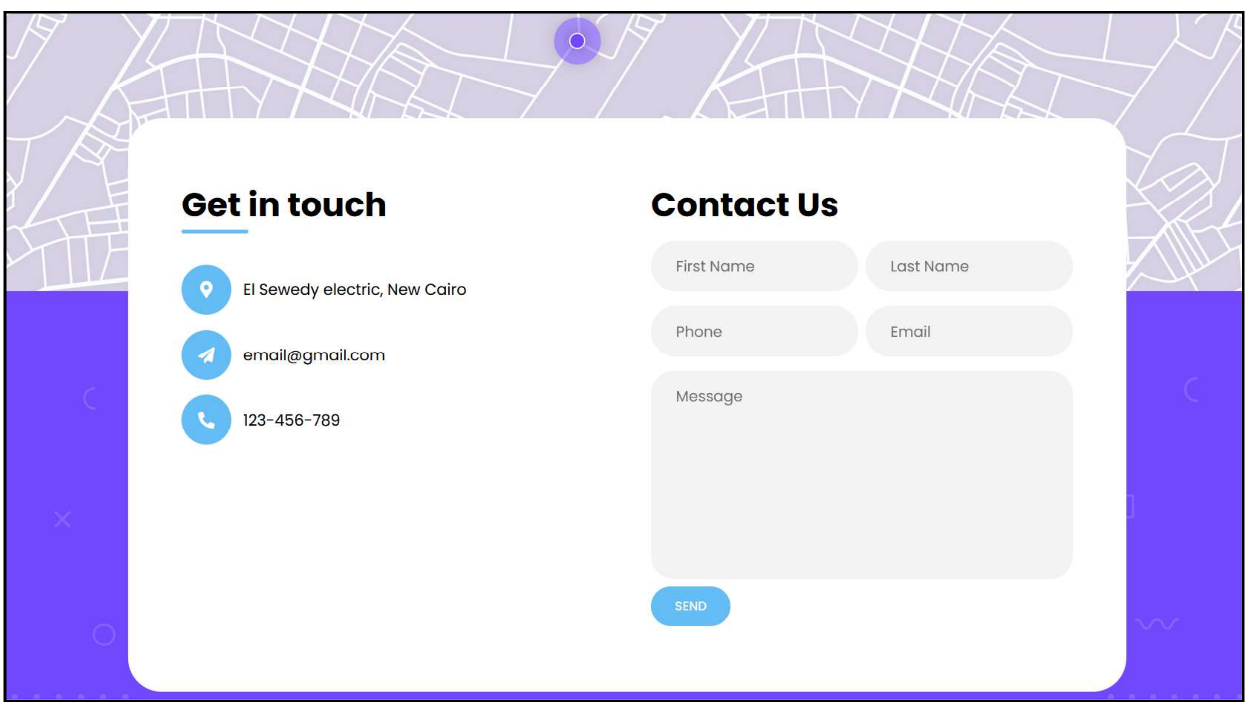


Figure 8.6: Contact us.

The last part in this page is a paragraph describing who we are, and the footer of the page including services, links and a button to let you go to the top of the page by clicking it as shown in **Fig 8.7**.

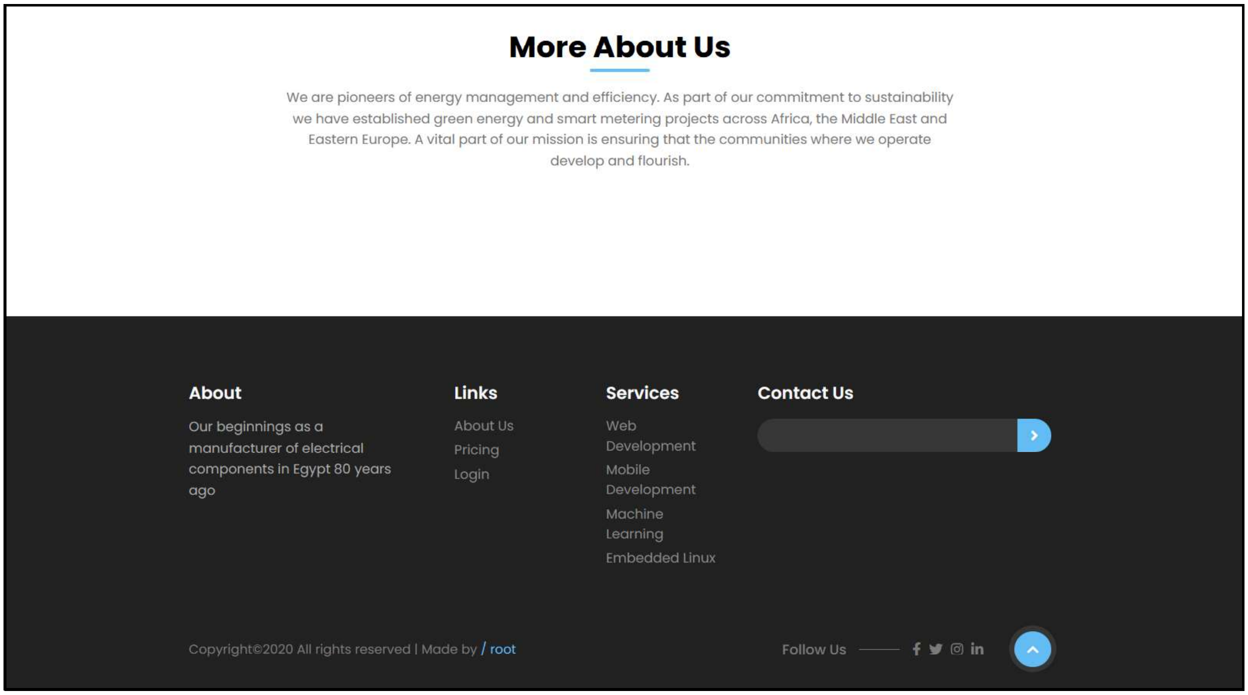


Figure 8.7: The footer.

- **About Us page:**

When clicking on the tab "About Us" in the nav bar the following Figure illustrates the page that will be opened.

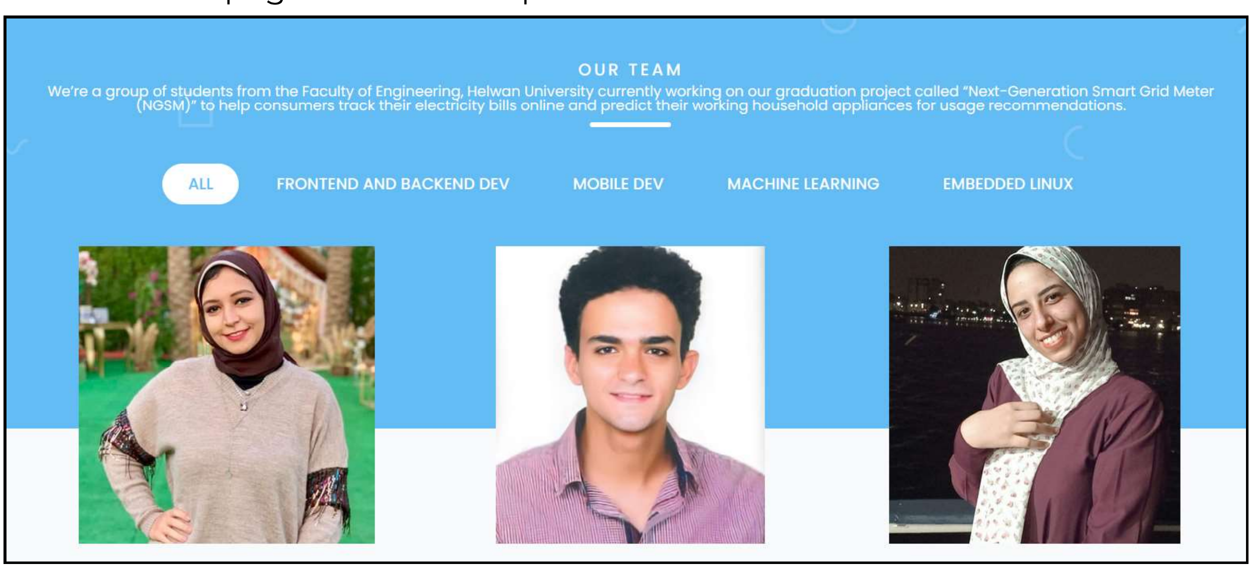


Figure 8.8: About us page

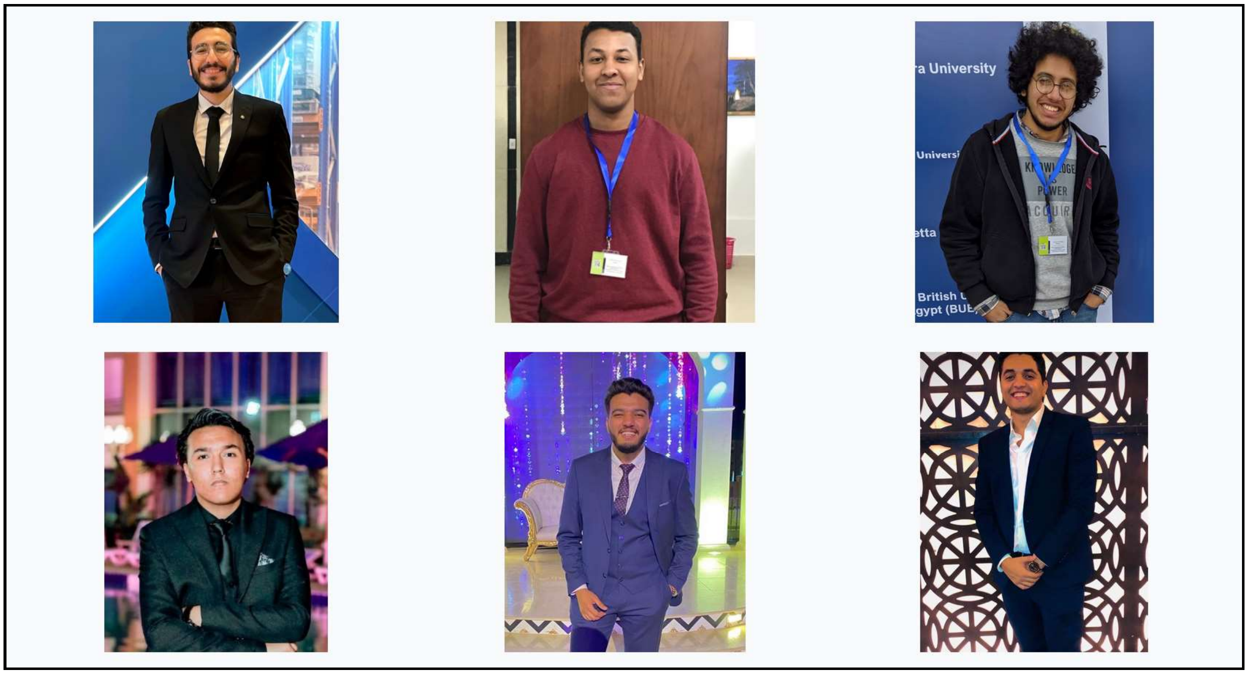


Figure 8.9: About us page

This page divides into several sections illustrating how we divide ourselves into groups to complete our project. We have a group for the web-based system as shown in the following figure.

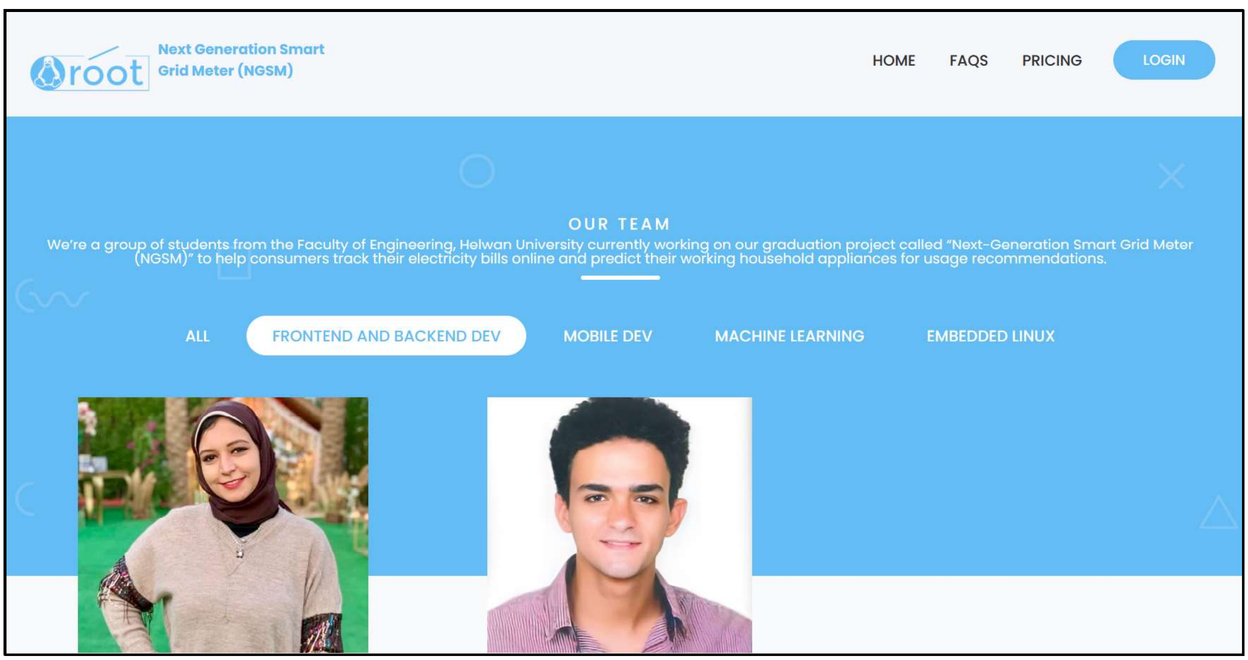


Figure 8.10: Web section page

Another section for the mobile development is as the follows:

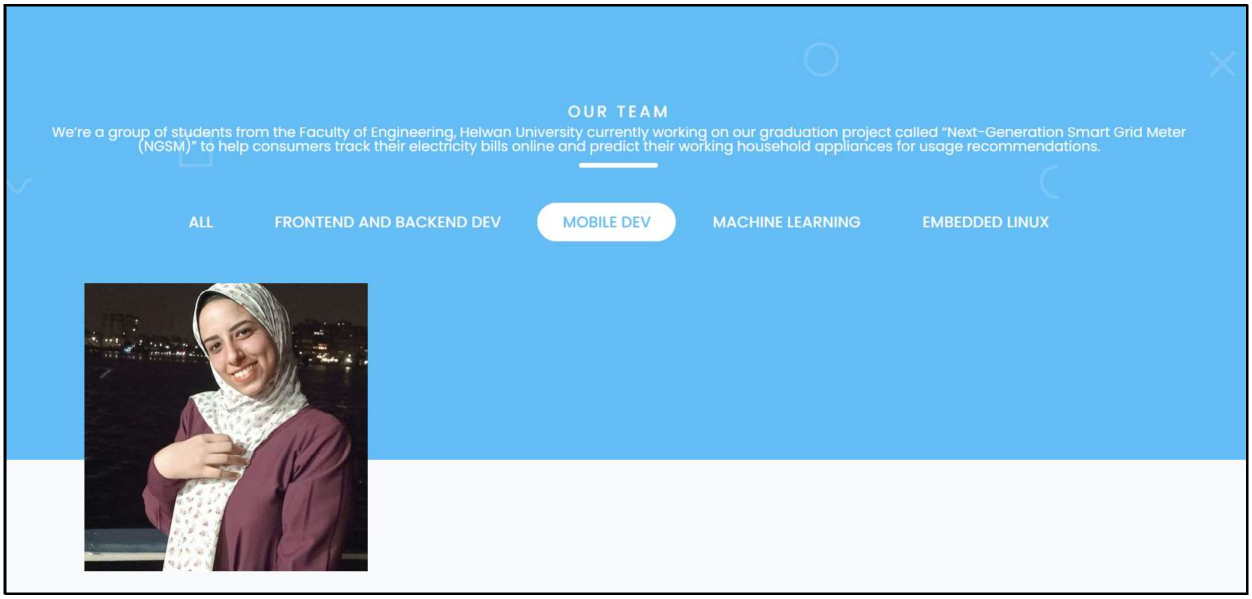


Figure 8.11: Mobile section page

Our team members who responsible for the machine learning section shown in the following figure:

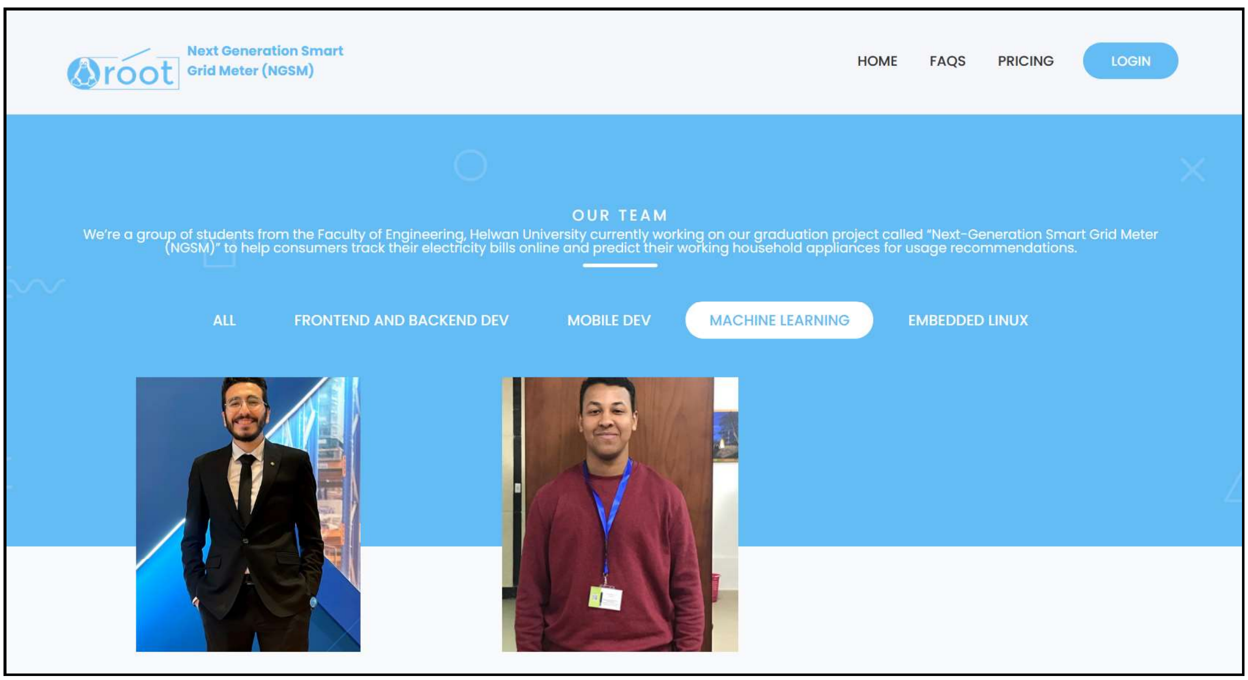


Figure 8.12: Machine Learning section page

Our team members who responsible for the embedded linux section shown in the following figure:

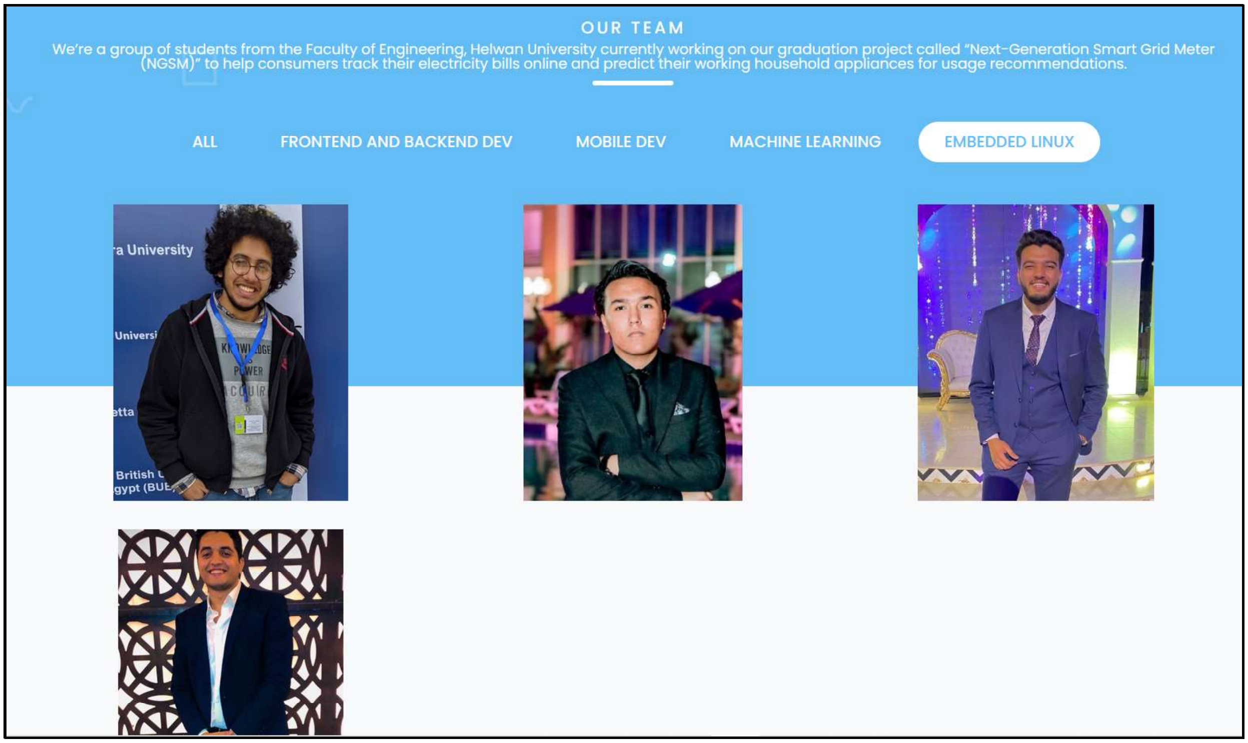


Figure 8.13: Embedded Linux section page

- FAQs page:

This page includes the most asked questions and their answers to help the user to know more about the system and our website.

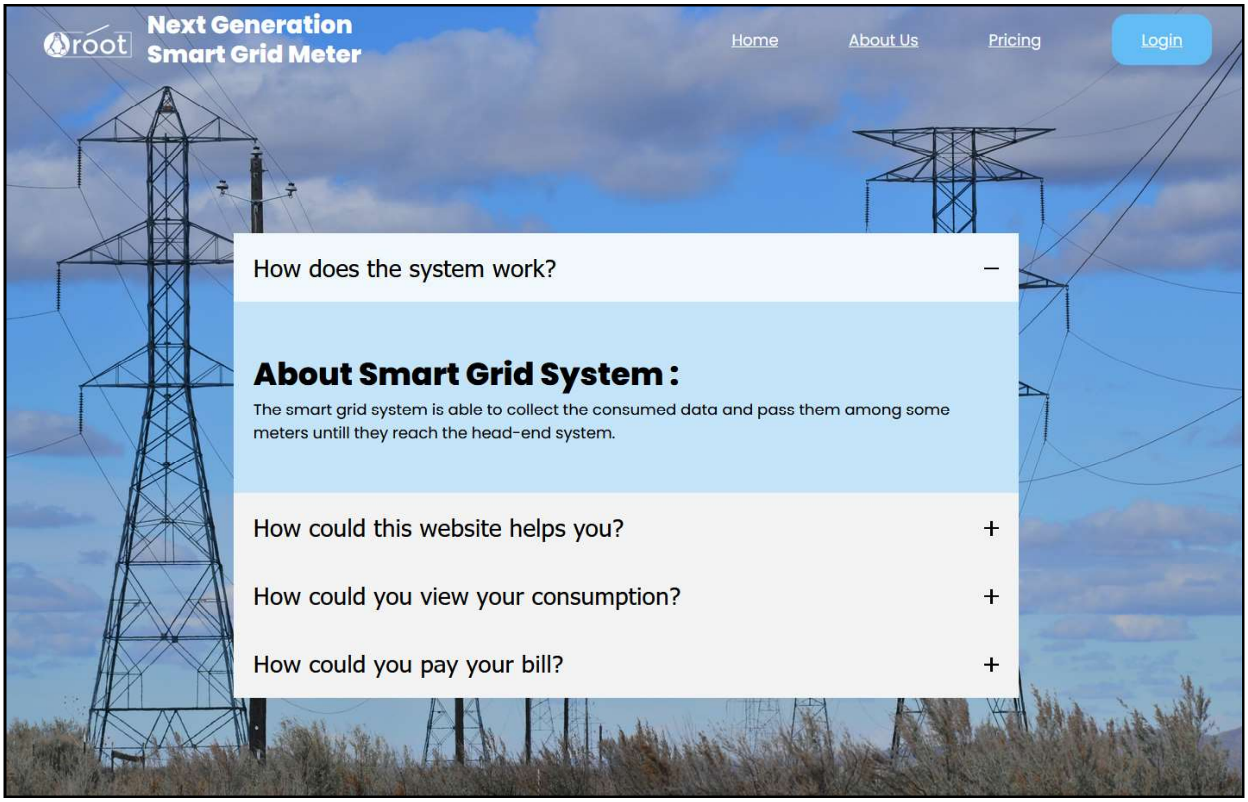


Figure 8.14: FAQs page

- **Pricing page:**

This page helps the user to know more about the prices and the offers in the company as shown in **Fig 8.15**.

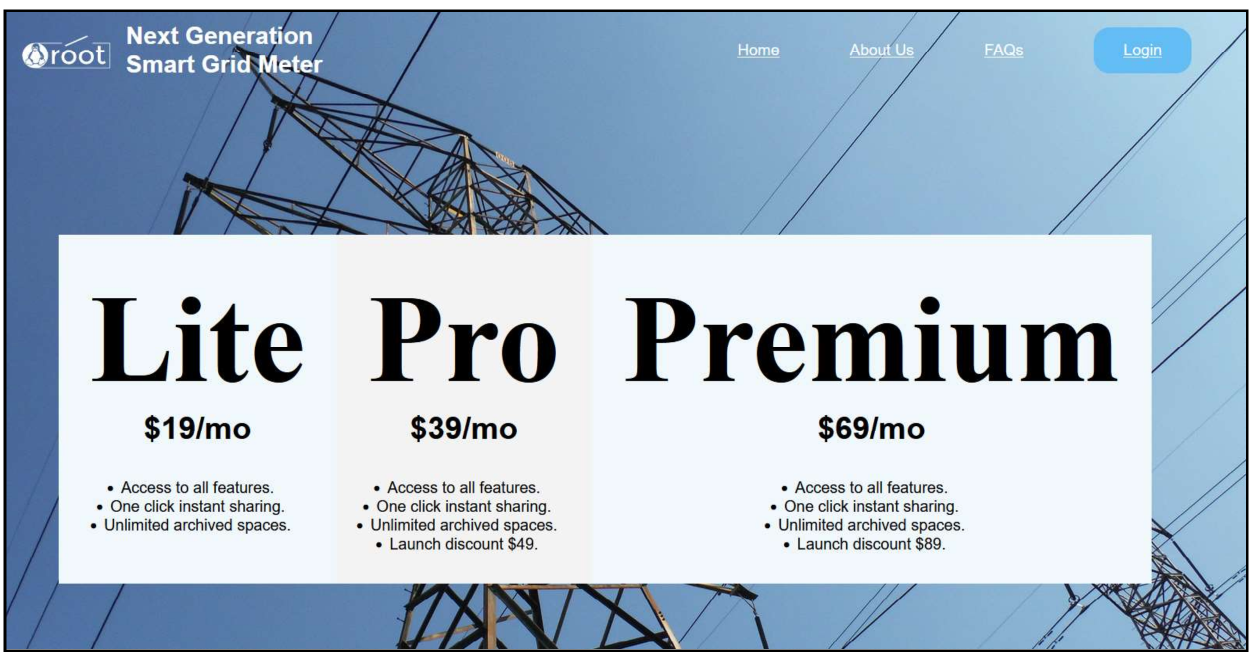


Figure 8.15: Pricing page

- **Login Form:**

This form cannot enable the user to login in until the username and the password both are correct. If the username contains one wrong letter and the password also, it will not enable me to enter. This form is totally secured.

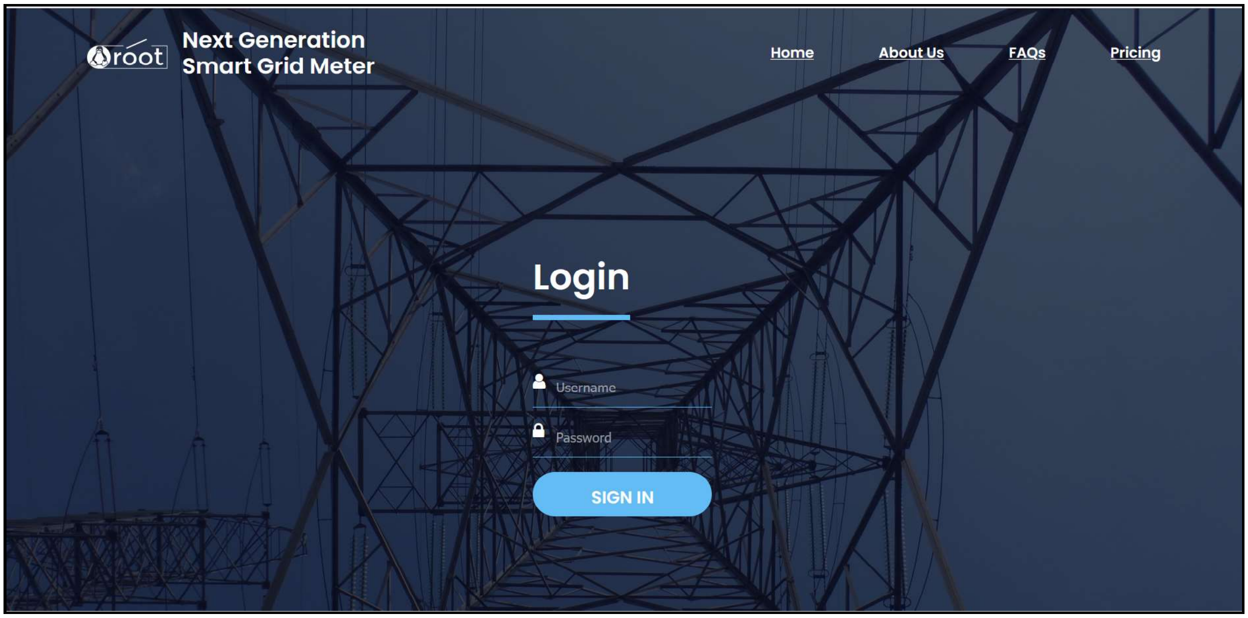


Figure 8.16: Login page

- **My profile page:**

After logging in, a page containing the user information appears. From this page, the user can choose any feature from the features shown in the following figure, even if viewing the consumptions, devices consumptions, viewing his bill and pay this bill online.

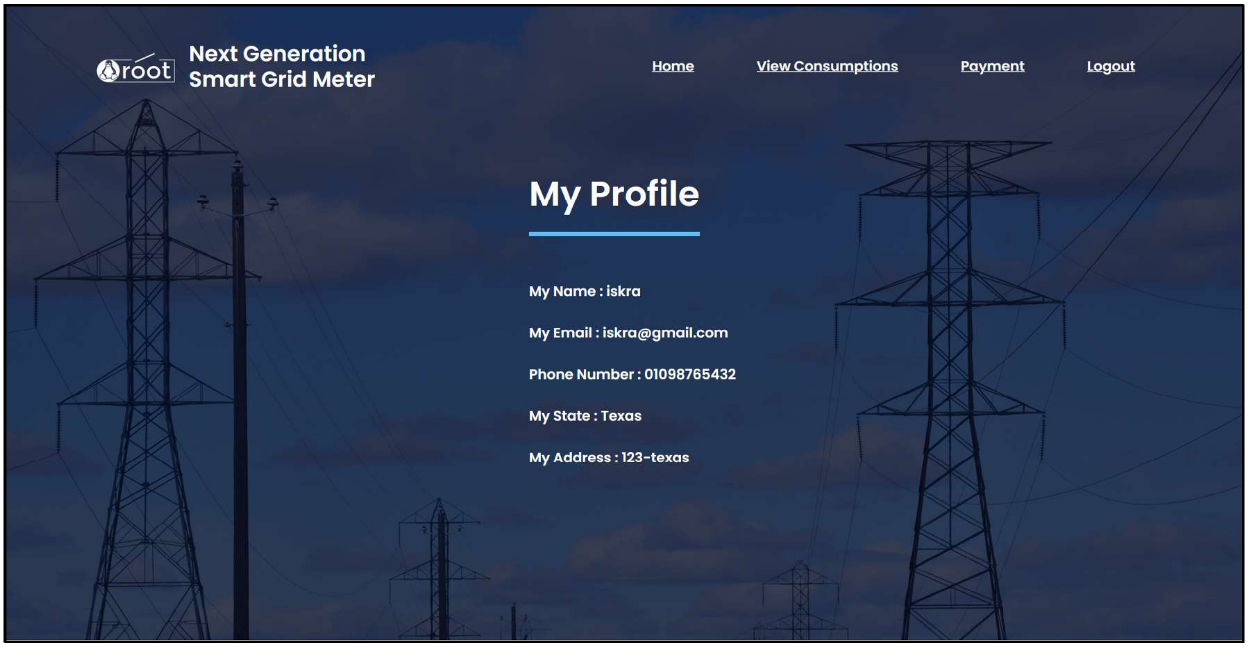


Figure 8.17: My profile page

- Consumption page :

This page enables the user to select the starting date and ending date, in order to view the consumption of energy as a form of bar chart to be an easy way for the user to predict the upcoming usage and to teach the user to reduce the consumption of energy. As shown in the following figure.

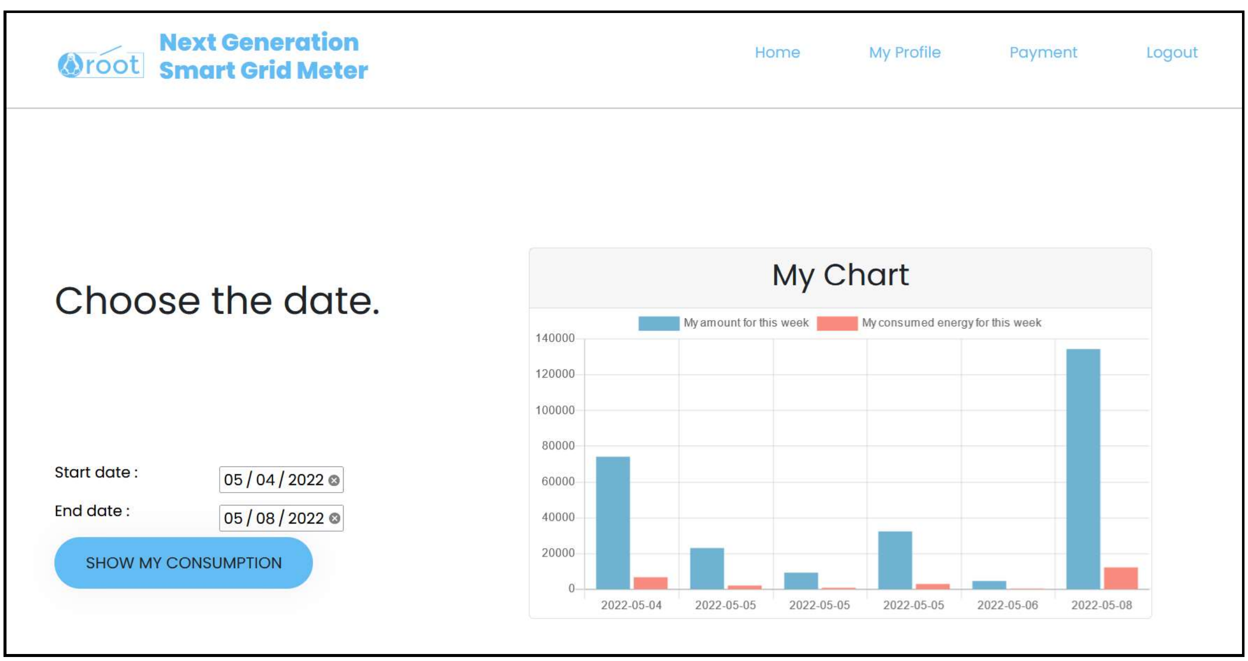


Figure 8.18: View Consumption page

- Payment

In this page, the user can view his bill and pay it online using his credit card. As shown in the following figure.

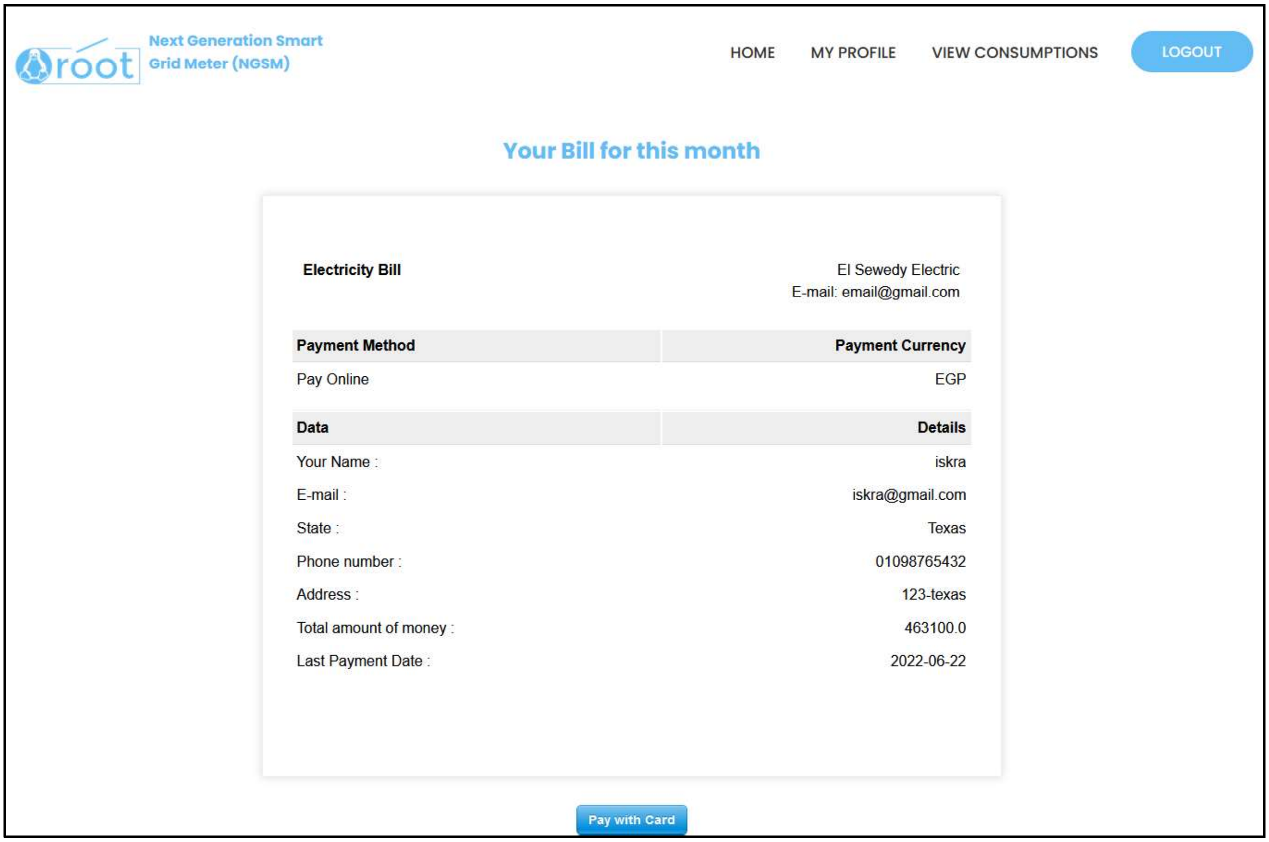


Figure 8.19: Bill page

After clicking the pay card button, the user has to enter the data about his credit card. If the user enters any wrong letters or numbers, the payment operation will not be completed as shown in **Fig 8.20.**

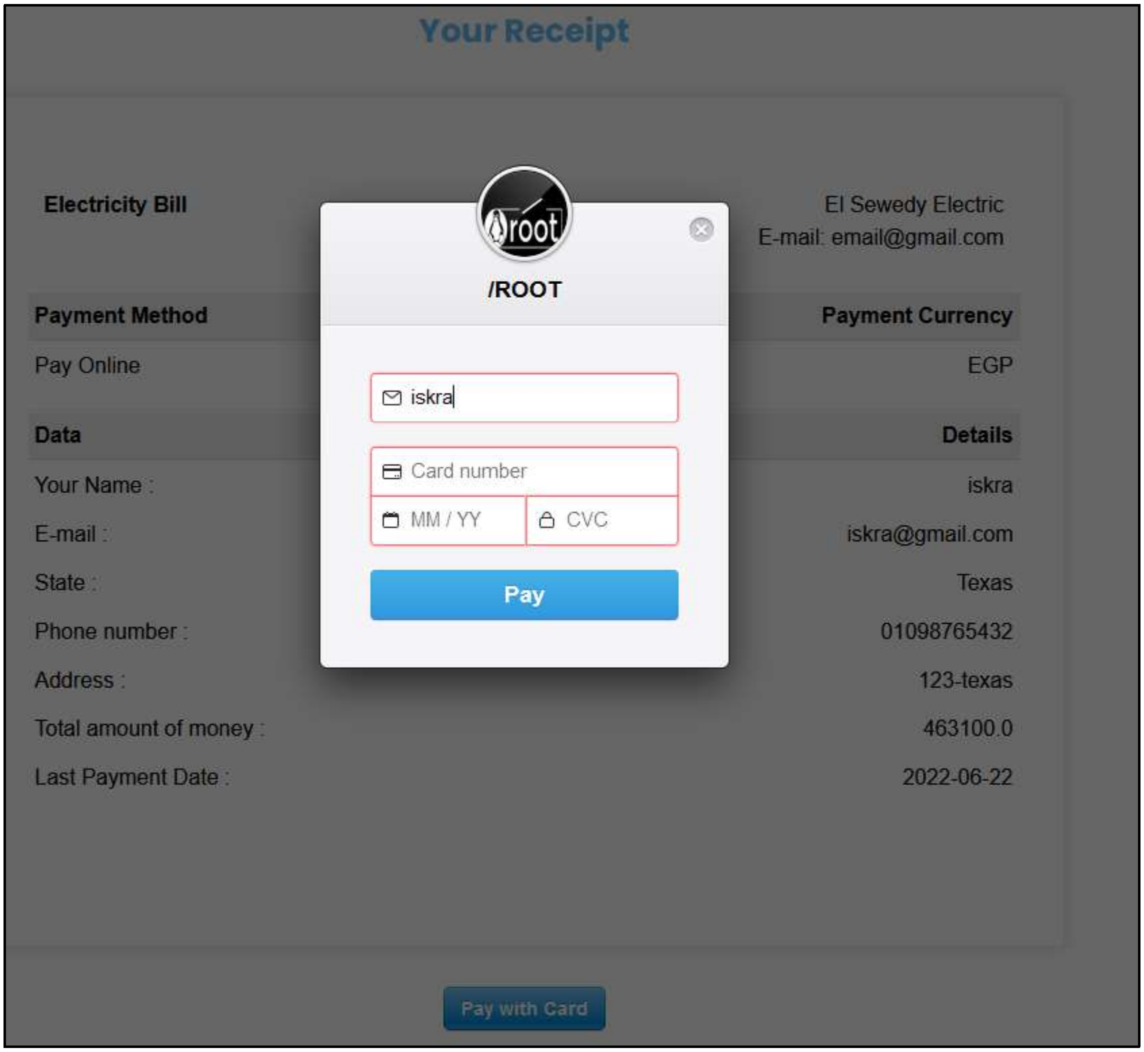


Figure 8.20: Payment page

These are the features of our system illustrated in detail with our GUI that is implemented in a simple way for the user to use it in an easy way.

9. NGSM Smartphone Mobile Application

9.1 Overview and Summary

An application called **"/root NGSM"** was created using the Flutter framework and Dart language. It was created to help users to communicate easily with the head end system and have some capabilities.

Using "/root", users have many benefits, like:

- Have an account.
- Show power consumption of any date.
- Show power cost of any date.
- Pay the bill online.
- Show old bills of old dates.
- Show each appliance consumption of power.
- Show which appliances are on and which ones are off.
- Know how to use the app in a form of FAQs.
- Contact with the head end system if it has a problem for example, users can do that via:
    - ☐ Phone call.
    - ☐ SMS.
    - ☐ Sending mail through Gmail.
- View the "About Us" page which contains some information about the whole system creators.
- Change your account settings, like:
    - ☐ Change username.
    - ☐ Change password.
    - ☐ Choose a profile picture.
    - ☐ Convert mode from light mode to dark mode, and vice versa.
    - ☐ Convert language from English to Arabic, and vice versa.
- Log out.

9.2 Mobile Application Block Diagram

In this section, the "NGSM Smartphone Mobile Application" is explained in a form of block diagram as shown in **Fig 9.1**. As it divides the system into a number of modules, each module presents some services and may deal with an API containing some data which helps to execute the services.

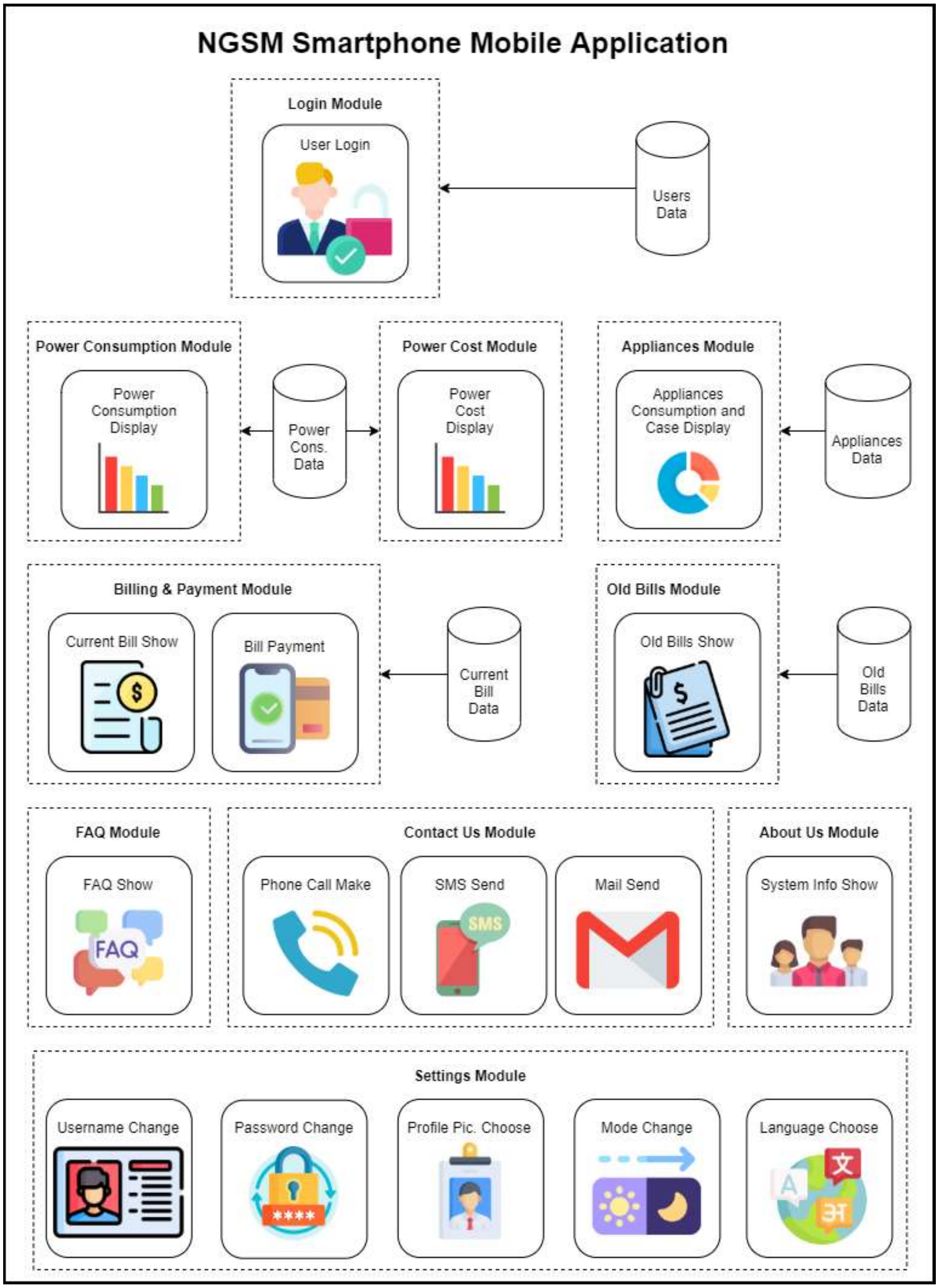


Figure 9.1: Mobile Application Block Diagram

9.3 Main Features

Now, going through the main features in details, they are as following:

- **Log in:** with this feature, a user can log into his account using email address and password. To log in successfully:
    - ☐ Correct email address and password should be entered.
    - ☐ Email address' text form field shouldn't be empty.
    - ☐ Password length shouldn't be less than 6.
- **Forget password:** as the user may forget his password, by then, he could enter his email address and reset his password.
- **Show power consumption and power cost:** To be able to do so, a start and end dates should be selected from the date picker in order to determine the range of dates you want to view.
- **Show bill:** If you open this tab, the bill will be viewed automatically.
- **Pay bill:** After showing the bill you can press the button "Pay Bill", this will launch a web page which helps you to pay your bill after asking you about some necessary information to finish the operation of payment successfully.
- **Show old bills:** To show an old bill which you have already paid, you should select a date from the date picker and then press the button "Show Bill" to show the bill that you have paid on the selected date.
- **Show each appliance power consumption:** In order to do that, you shouldn't have to do anything. As soon as you open this tab, you will find a circular progress indicator for each appliance, each one includes:
    - ☐ A picture of the appliance.
    - ☐ The name of the appliance.
    - ☐ The power consumed by the appliance.
    - ☐ The case of the appliance to know if it is ON or OFF.
- **FAQ:** This feature helps a user to understand well how to deal with the application, as it shows how to use each service in a form of question and answer.
- **Contact with the head end system:** To be able to communicate with us, three buttons are available:
    - ☐ The first one will enable you to make a phone call with us through a specific phone number.
    - ☐ The second one will enable a user to send us SMS, in which you could tell us whatever you want, through the same phone number.

- ☐ The third one will enable a user to send us mail through a specific email address on Gmail.

- **About Us:** This feature provides the user with a lot of information about us and the system, like: who we are and how we created the system.
- **Settings:** with this feature, user can change some settings of the account like the following:
    - ☐ **Change username:** by pressing the button "Change username", a new page will be opened to ask you to enter the new username you want, pressing the "Save" button will save the newly entered username but pressing the "Cancel" button will cancel the edit you made.
    - ☐ **Change password:** by pressing the button "Change password", a new page will be opened to ask you to enter the new password you want, pressing the "Save" button will save the newly entered password but pressing the "Cancel" button will cancel the edit you made.
    - ☐ **Choose a profile picture:** a user can do that by pressing the pen button attached to the profile picture. By then, he will be able to choose a profile from the gallery or take it by the camera.
    - ☐ **Choose mode of the app:** this feature helps the user to choose the mode of the application that he wants, light or dark, and this could be done through a switch button.
    - ☐ **Choose language:** by pressing the "Languages" button, a new box will appear, containing the available languages of the application to choose one of them.
- **Log out:** through the "Logout" button, the user will be able to log out his account on the application. Note that if the user didn't log out his account and turned off the application, his account will remain opened.

9.4 Grid Communication

The APIs used in the application will be discussed in this section, as we have used the following APIs:

- An API is used to login. As when the user wants to login, he enters his username and password and press the button "Login", By pressing this button, a request will be sent to this API, containing the entered username and password to make sure that they are correct, if so, the user will get a response with the user id and login successfully.

- The API used for showing power consumption is the same one used to show power cost of certain dates, the difference between them is just the key that will show the result.A user could select different start & end dates through the date pickers, these dates are put in this API, so, after pressing the button “Show Consumption” or “Show Cost”, the request will be sent to the previous API with the selected dates.
- An API is used to show the user bill is. As a request will be sent to this API containing the user’s id, then, the user will receive a response containing his bill.
- An API is used to show old bills. As a user can select a different date to send it to this API with the user’s id as a request after pressing the button “Show Bill”. After that, the user will get a response containing the bill he wants to show.
- An API is used to show the power consumption of each appliance, by sending a request to this API, the user will get a response containing each appliance name, consumed power and case to show if it’s ON or OFF.
- An API is used to change username. When sending a request to it, the user id and new username entered by the user should be attached.
- The API used to change Password is the same one used to change username. When sending a request to it, the user id and new password entered by the user should be attached. **[21]**

9.5 Procedure of Operation and Technical Demo

In this part, we will guide the user to know how to use the application:

- When the user opens the application, a splash screen will appear as shown in **Fig 9.2**.

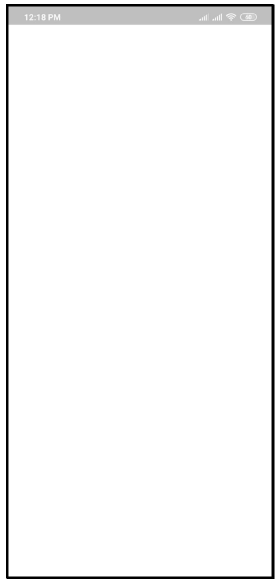


Figure 9.2: Splash Screen

- Then, the login screen will appear to ask the user to enter his email address and password and then press the “Sign in” button in order to log him into his account on the app as shown in **Fig 9.3**.

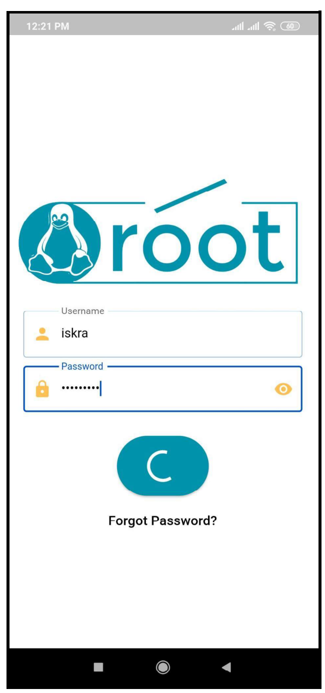


Figure 9.3: Login Screen

- If a user forgets his password, he can press “Forgot password?” button. A new page will appear to ask him to enter his email address to be able to reset his password as shown in **Fig 9.4**. From this page he can go back to the “Login” page by pressing “Go back to Login?” button.

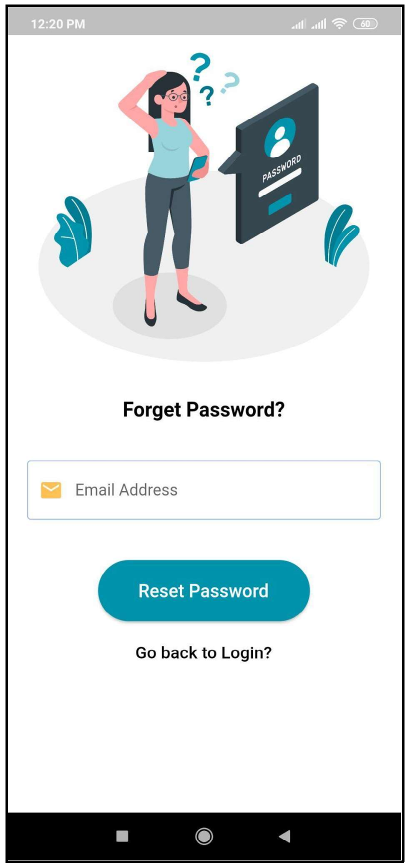


Figure 9.4: Forget Password Screen

- After that, the user can view his profile picture followed by his username, email address and the available services to him in a form of cards containing a picture expressing the service followed by the name of the service as shown in **Fig 9.5**.

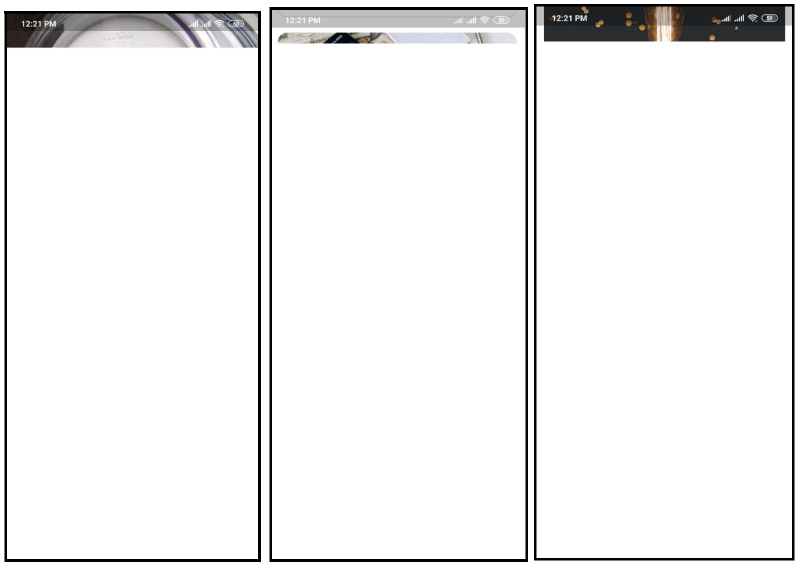


Figure 9.5: Profile Screen

- The First service is called "**Electricity Consumption**". This service consists of two tabs.
    - In the first tab "**Consumption**", the user should select the start & end dates of the period he wants through the date pickers, and then press the button "Show Consumption". As a result, a chart will appear after loading with a circular progress indicator. This chart shows the date on the horizontal axis and power consumed on the vertical one as shown in **Fig 9.6**.

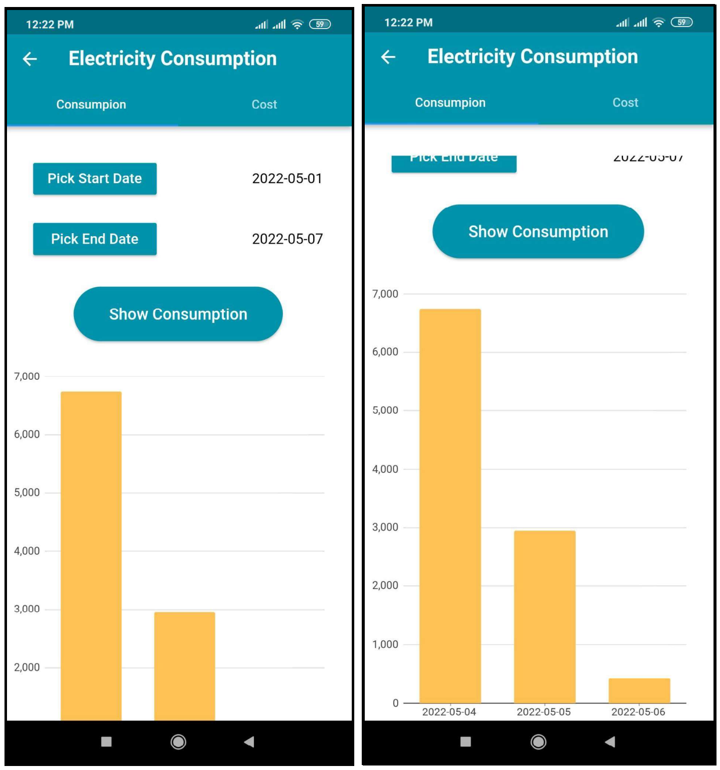


Figure 9.6 Consumption Tab

☐ In the second tab "**Cost**", the user should do the same, as he should select start & end dates through the data pickers and then press the button "Show Cost". As a result, a chart will appear after loading with a circular progress indicator too but this chart shows the date on the horizontal axis and power cost on the vertical one as shown in **Fig 9.7**.

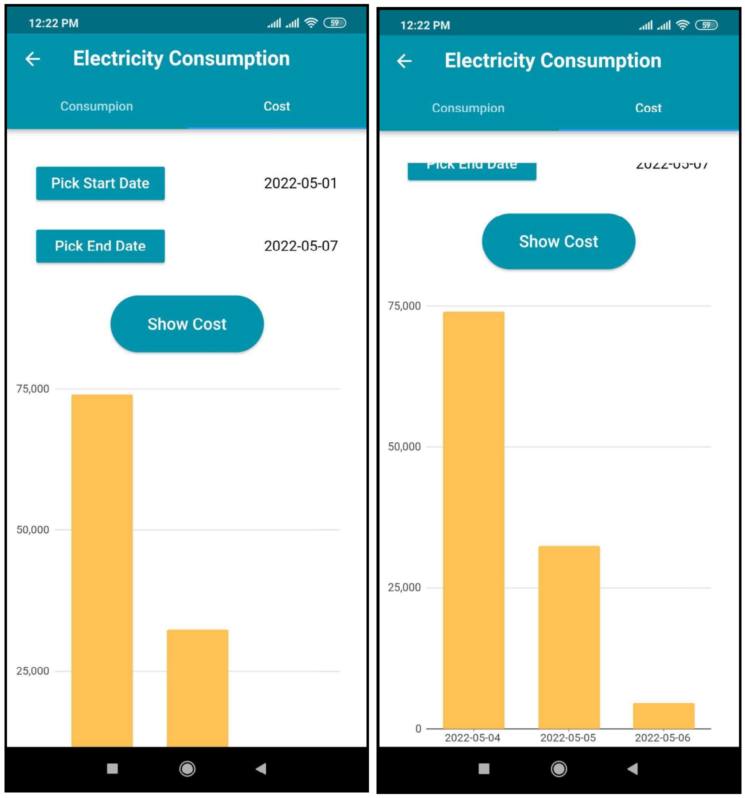


Figure 9.7 Cost Tab

- The second service is called "**Payment & Bills**". This service consists of two tabs.
    - ☐ In the first tab "**Pay Bill**", User can view his bill after loading with a circular progress indicator. After that, if the user wants to pay his bill, he should press the button "Pay Bill" as shown in **Fig 9.8**. As a result, a web page will be launched to ask the user to enter some data in order to enable him to finish the payment operation.

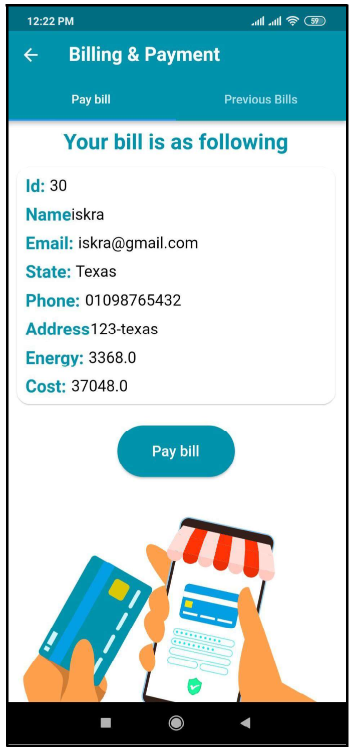


Figure 9.8 Pay Bill Tab

☐ In the second tab "**Previous Bills**", a user can find his old bills easily. To do so, a user should select the date he wants and then press the button "Show Bill". As a result, the old bill that was paid on the selected date will appear after loading with a circular progress indicator as shown in **Fig 9.9**.

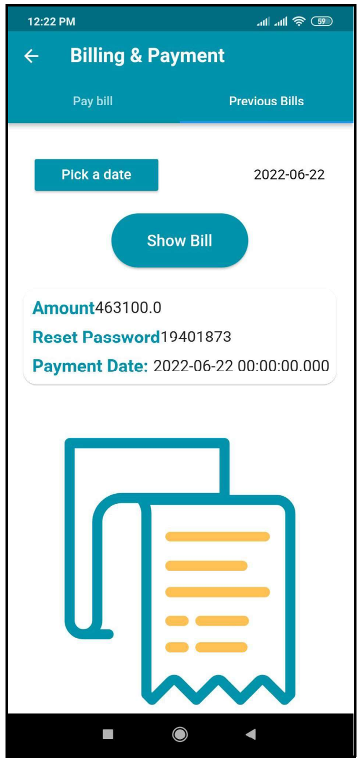


Figure 9.9 Previous Bills Tab

- The third service is called “**Appliances**”, the user can view the power consumption of each appliance in his house like fridge and kettle after loading with a lottie file, each appliance consumption will be expressed in a form of a circular slider as shown in **Fig 9.10**.

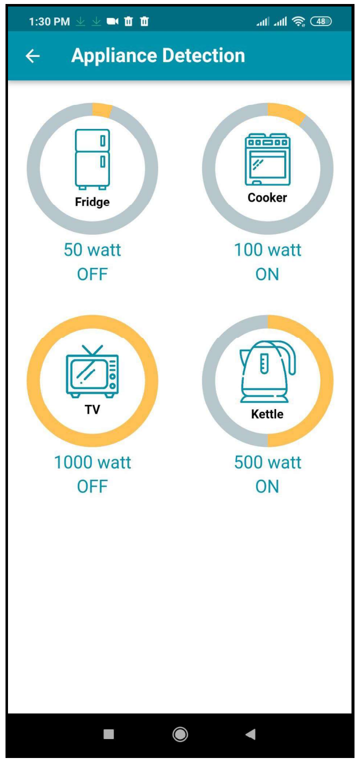


Figure 9.10 Appliance Detection

- The fourth service is called "**Frequently Asked Questions**", the user can use this service to know how to deal with other services as it shows that in the form of questions & answers as shown in **Fig 9.11**.

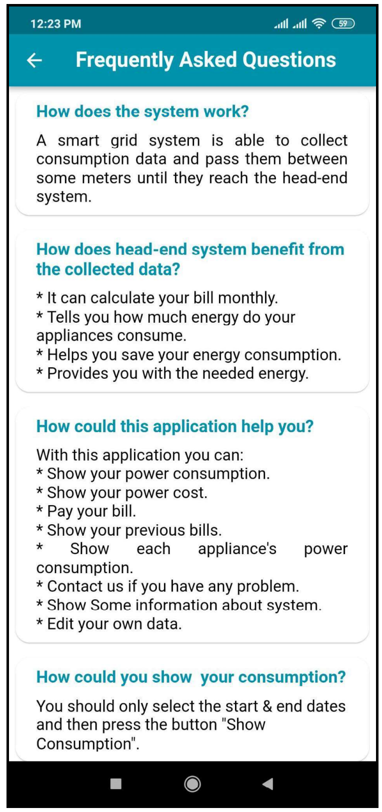


Figure 9.11 FAQs

- The fifth service the user can use is called "**Contact us**", by pressing on this service, the user will find three options to choose one from to communicate with us if he is facing a problem as shown in **Fig 9.12**. These tree options could be:
  - ☐ The first option is called "Via Phone Call". By choosing this option, a user will have the ability to call us through a fixed number.
  - ☐ The second option is called "Via SMS". By choosing this option, a user will have the ability to send us SMS through the same number.
  - ☐ The third option is called "Via Mail". By choosing this option, a user will have the ability to send us mail through a fixed email address.

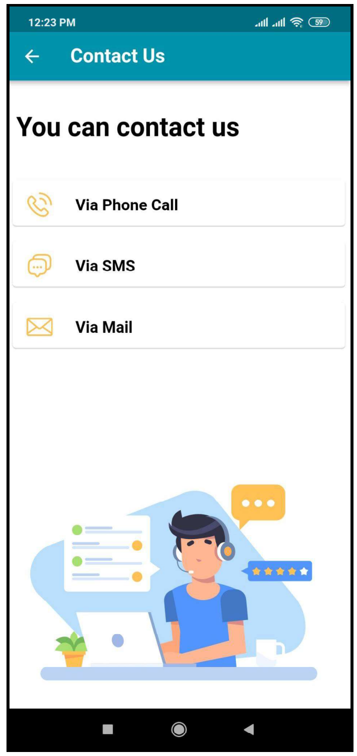


Figure 9.12 Contact Us

● The sixth service of the application is called "**About us**", by pressing on this service, the user will have the ability to collect some information about us and about the system as shown in **Fig 9.13**.

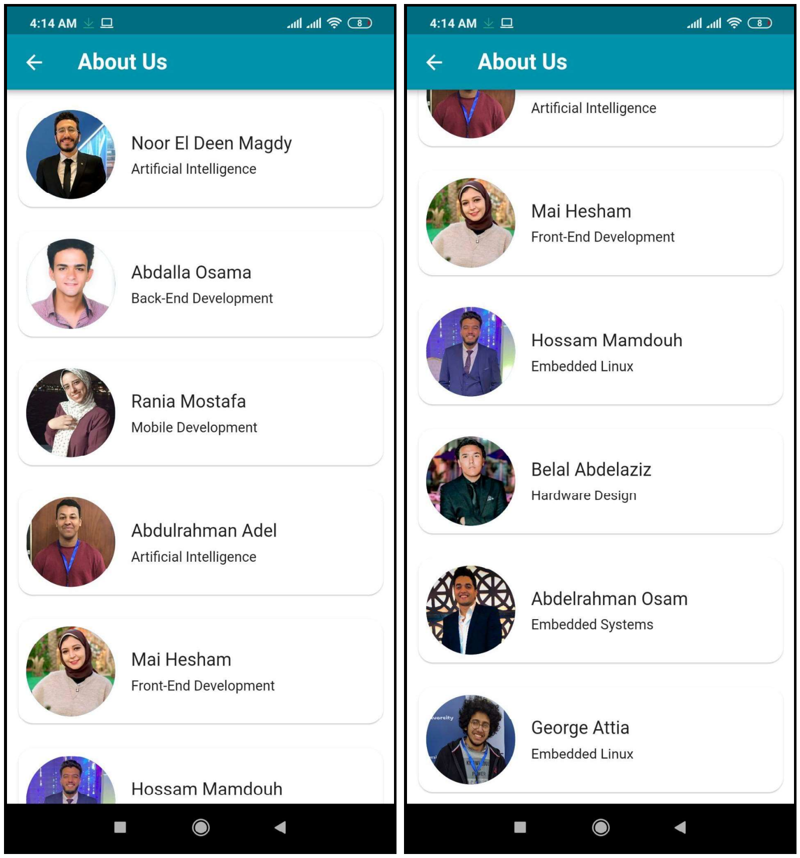


Figure 9.13 About Us

- The seventh service is called "**Settings**", through this service user will have some capabilities which are divided into two sections:

  The first section is called "**General Settings**" and it contains the following capabilities:

  - ☐ Change the language of the application through the button "**Languages**". By pressing this button, a box containing the available languages will appear to select the wanted language from.

    Note that only Arabic & English languages are available for now.

  - ☐ Convert the application mode from light to dark through the switch button as shown in **Fig 9.14**. A user can switch it to the light mode again.

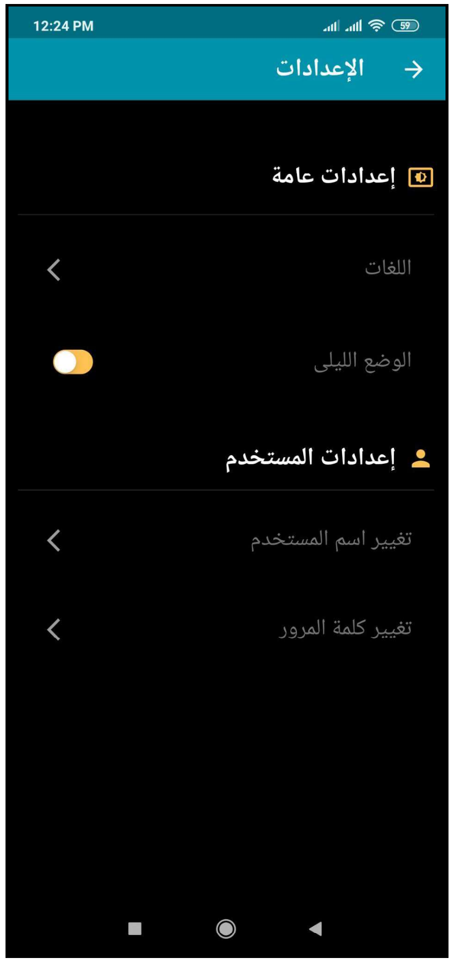


Figure 9.14 Settings

The second section is called "**User Settings**" and it contains the following capabilities:

- ☐ Change the username by entering the new username the user wants to convert its username to as shown in **Fig 9.15**, through the option "**Edit Username**". Then pressing the button "**Save**" will save the edit but pressing the button "**Cancel**" will cancel it.

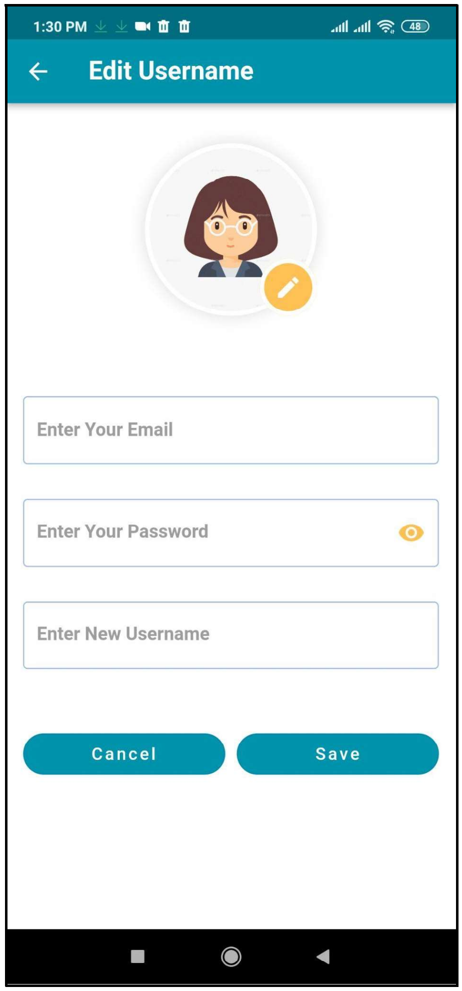


Figure 9.15 Edit Username

☐ Change the password by entering the new password the user wants to convert its password to as shown in **Fig 9.16**, through the option "**Edit Password**". Then pressing the button "**Save**" will save the edit but pressing the button "**Cancel**" will cancel it.

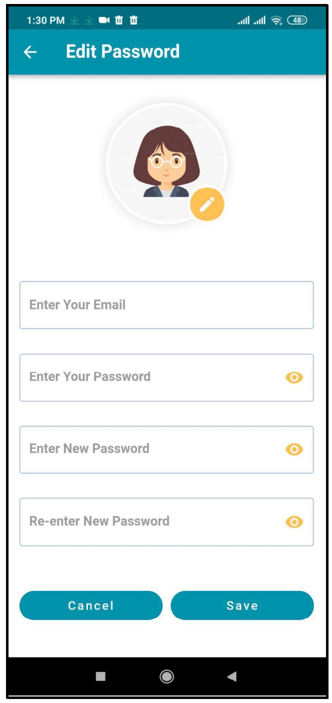


Figure 9.16 Edit Password

Note that through both options "Edit Username" and "Edit Password", the user can change his profile picture too through the pen attached to the profile picture, as it will enable the user to take a photo through the camera of the mobile phone or choose a picture from the gallery.

- A "**Logout**" button is following the previous services. Pressing on it will log the user out and redirect him to the "Login" page again.

  Note that all services have a "**Back**" button in their "AppBar" which leads to the user's profile picture again.

10. NGSM Appliance Detection and Electricity Forecasting

10.1 NILM Background

How to Recognize Appliances from Smart Meter Data using AI?

Identification of electrical appliances from Smart Meter Data (total consumption of a household) is called Energy Disaggregation or Non-Intrusive Load Monitoring (NILM).

Instead of measuring every appliance in a household (also called Intrusive Load Monitoring or ILM), we can use machine learning to analyze the total energy consumption and determine the energy consumption signals originating from electrical appliances.

The major advantage of NILM over ILM is that you only need one Smart Meter in a household instead of many devices each measuring desired electrical appliance.

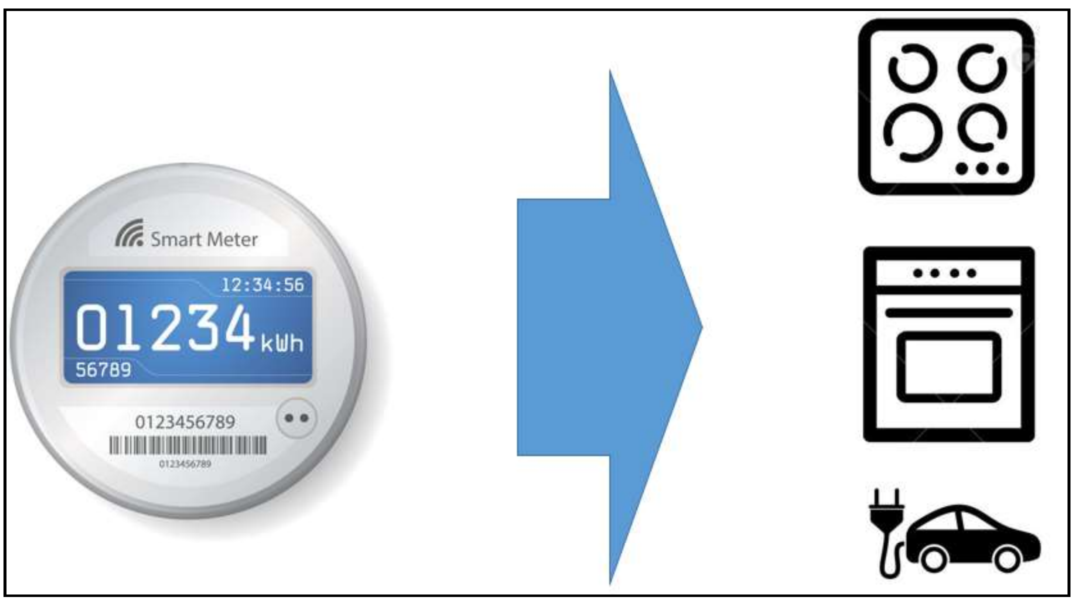


Figure 10.1: Non-intrusive load monitoring (energy disaggregation) framework.

Why do we need NILM?

ILM or NILM can be used in the scope of: energy management systems, network planning, and operation, or demand-side management (eg. targeting consumers in energy efficiency or demand response programs).
Products based on NILM technology are usually advertised that they can help reduce energy consumption via feedback. The idea is to inform consumers about the energy consumption of major appliances in the household in order for them to save electricity. If you know which devices consume the most, you can reduce (or shift their usage).

How does NILM look in practice?

The existing Smart Meter infrastructure is not appropriate for NILM.

First I have to emphasize that the existing Smart Meter infrastructure that usually uses 15-min, 30-min, or 60-min resolution is not appropriate for NILM as we need higher frequency data such as 10s or 1s or even higher for the algorithms to work properly.
There were many start-ups founded in recent years leveraging NILM and claiming magical results. Usually, consumers have to buy a device (Smart Meter) that measures whole-home electricity in a high resolution. The application further sends the data in the cloud where machine learning algorithms are applied and finally shows the disaggregated consumption in a fancy mobile app.

How to model NILM?
In recent years, deep learning-based approaches have become very popular for solving NILM.
There are many approaches, whereas I will provide a short description of the (probably) most popular approach which is also quite self intuitive.

Let's say you want to develop an algorithm that is able to recognize three electrical appliances (e.g. dishwasher, oven, fridge). You have active power (in Watts) time-series data from smart meter readings (also called mains), which represents a whole-home energy demand and is an aggregate of all the devices in a household (denoted P_mains). Additionally, you have three-time series measurements for each device separately (the active power of a dishwasher is denoted as P_dishwasher).

The main goal is to develop a model that takes smart meter readings (P_mains) as an input and outputs disaggregated electrical signals of three devices (P_dishwasher, P_oven & P_fridge).
You have to develop one model for each device separately:

P_dishwasher = f (P_mains)

And analogically for the other devices.
Further, you use deep learning to estimate f.

This is a regression problem, where data samples are created using the rolling window approach (in our later case each window had a length of 361).
There are two main approaches for modeling f:

sequence to point (model input is a sequence of data and output is a single point)

sequence to sequence (model input is a sequence of data and output is also a sequence).

Deep learning (especially deep convolutional neural networks) methods perform very well for NILM. The biggest issue with NILM is that publicly available datasets have measurements only for a few houses.
This is probably not enough for training algorithms that would ideally generalize very well to a thousand or millions of houses not seen during training.

In the Figure below (top graph) we can see whole-home energy from smart meters colored by the energy consumption of each device. On the bottom graph, there is an example of using NILM for recognizing dishwashers. Green represents the actual signal of the device and red the predicted signal. We used deep convolutional neural networks (sequence to point model with 1D convolutions and VGG blocks) for encoding the input signal and a fully-connected layer as a decoder on a publicly available dataset (REDD) with 10s resolution. We learned one model per device, whereas the models were able to recognize around 90 % of individual device energy. **[28]**

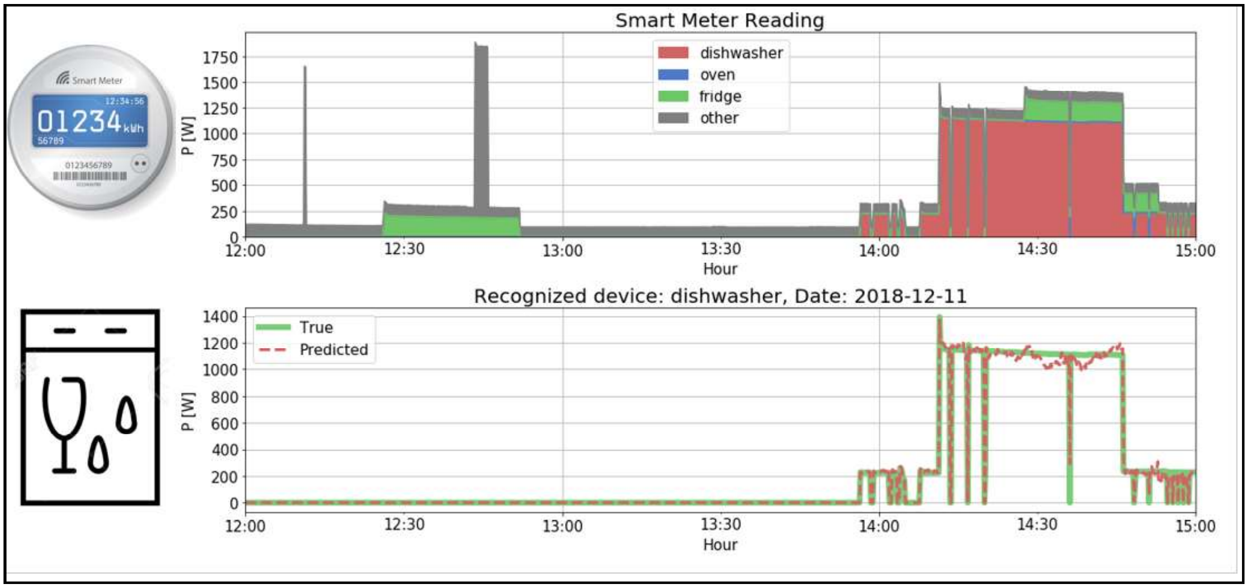


Figure 10.2: Results of using Deep Learning for NILM on REDD dataset.

Conclusion

NILM is a very interesting research area, especially these days when new deep learning approaches are developed almost on a daily basis. Nevertheless, due to the lack of publicly available datasets for a large number of households, it is hard to assess the practical value of these new approaches as you can not evaluate your algorithms on a wide range of different households. Therefore, you cannot properly assess whether these algorithms would work well in real-world environments and if wide adoption of NILM would even work.

10.2 Previous Approaches

Energy disaggregation systems allow to obtain information relating to the power absorption of individual appliances connected to a user, through the use of voltage and current transducers positioned at its connection point to the grid

There are two broad approaches to disaggregation: optimisation ("non-event-based NILM") and pattern recognition ("event-based NILM").

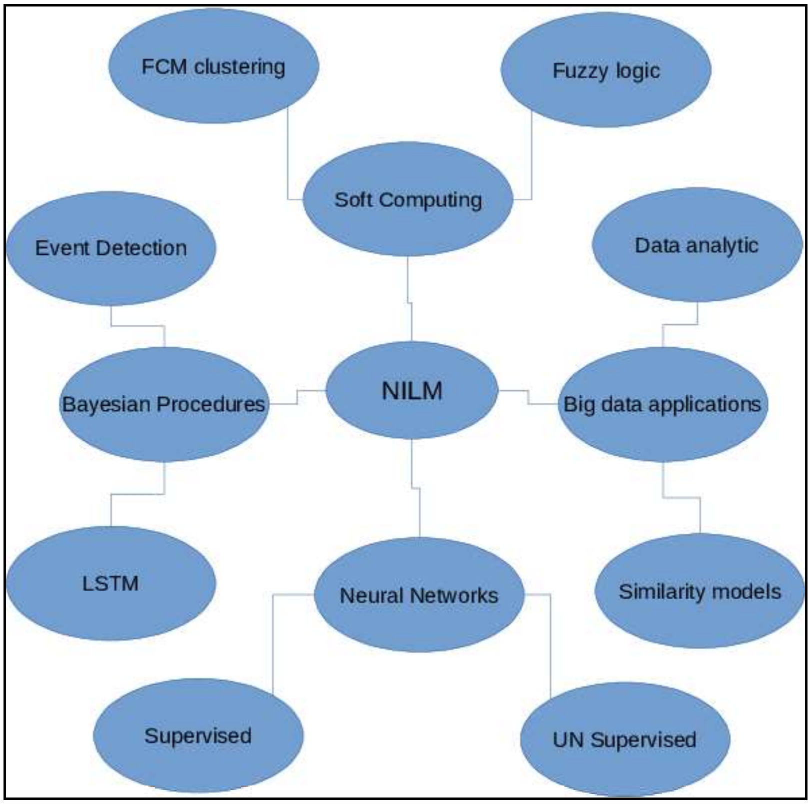


Figure 10.3: State-of-the-art Energy disaggregation techniques

- **Disaggregation framed as an optimization problem**

The optimization problem can be described as follows: a smart meter signal is a time-series Y = {y1, y2, ..., yk} where k is the number of samples and yt is the meter reading at time t. We describe the state of each appliance with a boolean vector A = {a1, a2, ..., an} for n appliances. If appliance i is on then ai = 1; else ai = 0. The power consumption of each appliance is described in vector P = {p1, p2, ..., pn}. For example, if appliance 1 is a fridge which uses 100 watts and is currently on then a1 = 1 and p1 = 100. The total power consumption yt at time t is the sum of the power consumption of all active appliances:

$$y_t = \sum_{i=1}^{n} a_i p_i + e_t$$

Where et is an error term.

If the power consumption of every appliance is known then disaggregation can be stated as a combinatorial optimisation problem where, for every time slice t we try to find a state vector At such that:

$$A_t^* = arg\ _A min \left| y_t - \sum_{i=1}^{n} a_i p_i \right|$$

George Hart, one of the early pioneers of disaggregation research, points out that the optimization problem specified in this equation is an NP-complete “weighted set” problem and that a precise solution is only achievable by enumerating every possible state (G. W. Hart 1992). This is computationally impractical because n appliances, each of which can occupy any one of s states, can be configured in sn combinations so the computational complexity blows up exponentially as O(sn).

- **Hart’s non-intrusive load monitoring algorithm**

The first NILM system was proposed by G. W. Hart in 1985 , consisting of an algorithm based on the identification of events through edge detection and clustering.

Hart describes a ‘signature taxonomy’ of features which might be useful for distinguishing between appliances. His earliest work from 1984 described experiments of extracting more detailed features. However, Hart subsequently decided to focus on extracting only transitions between steady-states.

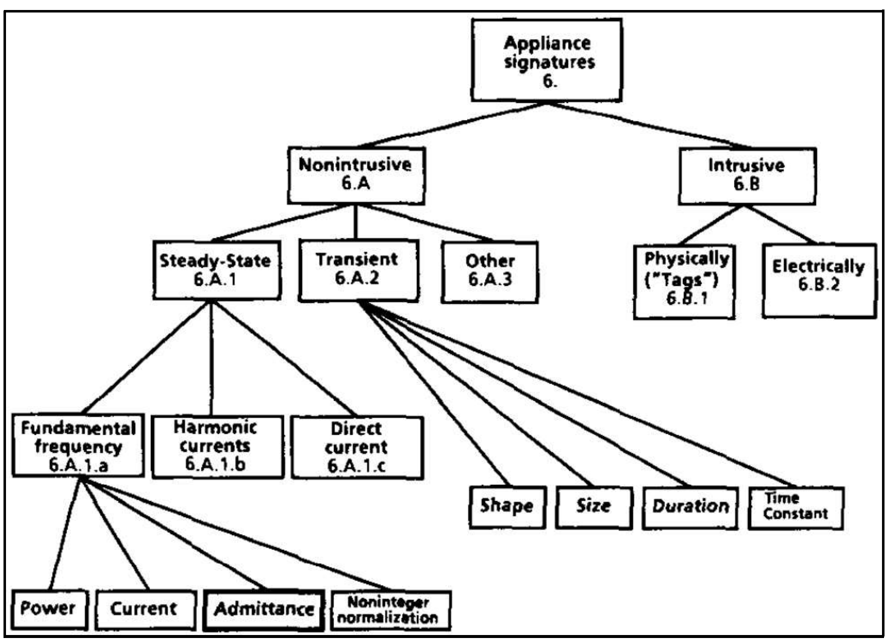


Figure 10.4: George Hart’s ‘signature taxonomy’.

Many NILM algorithms designed for low frequency data (1 Hz or slower) follow Hart’s lead and only extract a small number of features. In contrast, in high frequency NILM (sampling at kHz or MHz), there are numerous examples in the literature of manually engineering rich feature extractors

Hart proposed a classification of appliances based on their operating states:

1. ON/OFF appliances, e.g. lamps, toaster, etc.
2. Multi-state appliances, with a finite number of operational states, also called finite state machines (FSM).
3. Continuous variable appliances, i.e. appliances that absorb a variable amount of power without a fixed number of states, such as the drill.
4. Permanent consumer appliances, i.e. appliances that remain active for weeks or days absorbing power at a constant rate.

Clearly, his approach, although it was functional in certain situations, showed significant limitations since a multi-state appliance had to be managed as a set of distinct ON/OFF appliances, while the continuous variable appliances and the permanent consumer appliances could not be detected correctly. Since then, many researchers have tried to provide better solutions to Hart's one, both by measuring different quantities in addition to the active power and by using different algorithms.

- **Deep Learning**

Over the years, the use of deep learning algorithms has allowed us to overcome many of the limits that have characterized previous methods, thus allowing measurement systems to adapt to homes never seen during the training phase. The most successfully employed neural network algorithms are Recurrent neural networks and Long Short Term Memory. There also exists an RNN based approach for NILM on small power office equipment

- **DAE**

Autoencoders are made of simplistic neural structures and share similarities with principal component analysis. If an autoencoder is composed of a linear activation within each of its layers, it is highly likely that the latent variables present at the bottleneck of the encoder network (the smallest layer in the network) might directly resemble the principal components from PCA. They basically are a set of unsupervised machine learning concepts that result in reducing the inherent dimensionality of the projected data while preserving its essential features and enable the data from a higher dimension to be synthesized in a lower dimension using non-linear transformation. It comprises two functions, specifically, encoder and decoder. The process of data compression and down-sampling the input-data into a fewer number of bits and mapping it back to the latent space is performed by the encoder. The decoder function then puts the input encoding to use, reconstructing the input and mapping it to the latent space. The latent space is present at the bottleneck. A special deep neural network design implemented to extract a particular input component from the received noisy data is termed a **denoising autoencoder.** As the name suggests, it subtracts the noise from a given input in order to extract meaningful data. It forces the network to undergo overfitting on the random noise by adding some amount of white noise to the input. The error incurred is then compared to produce the required output. Removal of grain from input images and the reverb of voice signals are some popular applications of DAE. For its usage in NILM, the mains signal has been treated as a noisy appliance-power signal and the mains signal is considered to be the summation of the total power consumed by the target appliance and some additional noise. Since DAE denoises one appliance at a time, it needs multiple trained models for a group of appliances to be disaggregated. Moreover, in this scenario, the DAE gets as input a window of the mains readings (length of time-window remains fixed) and produces the induced appliance consumption values for the same time window as output.

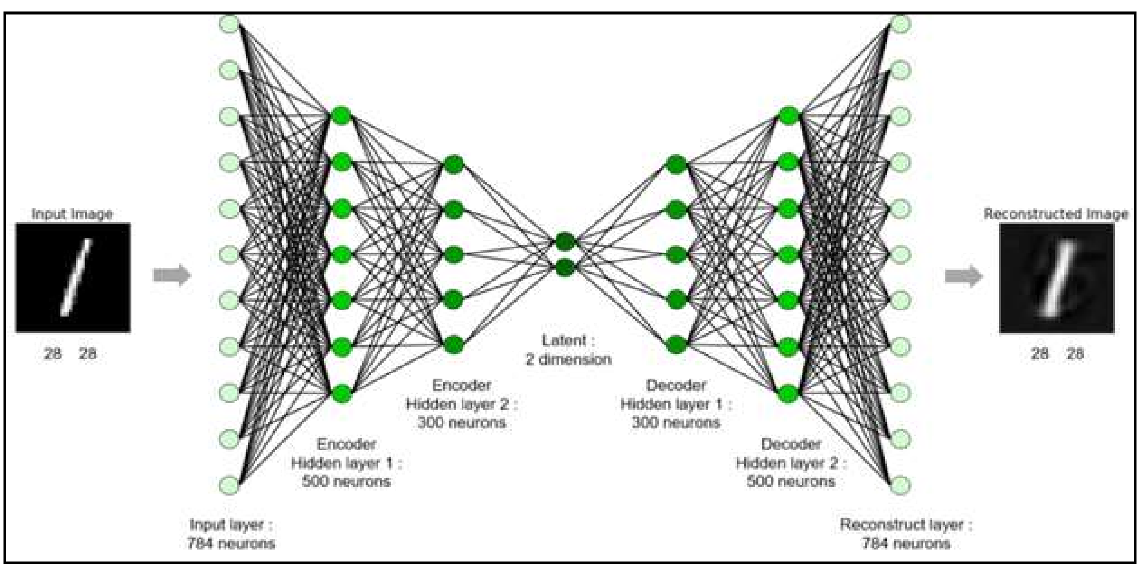


Figure 10.5: DAE

- **Sequence-to-Sequence**

The sequence to sequence learning model is a deep learning concept that is used to convert from one sequence to another. It contains an encoder RNN to understand the input sequence and a decoder RNN to decode the thought vector thereby constructing an output sequence. The process of condensing the input into a vector is achieved by an encoder network which passes its output to the decoder network where the vector is unfolded to form a new sequence. There's a set of RNNs that operate commonly between the encoder and the decoder in a sequence-to-sequence model. At each step of recurrence in the encoder network, a new word is fed to the input which gets utilized in the subsequent step by the next state. Once the decoder receives the final state from the encoder, it applies a discrete probability distribution (to the input at each step) to predict the output taking into consideration a loss function. The model here tries to map the sequence of mains reading to the target appliance sequence by learning a regressive map between them. The seq2seq model uses a regression expression which is defined by xt : t + W/1 = f(Yt : t + W/1, Q) + Et where f is a neural network.

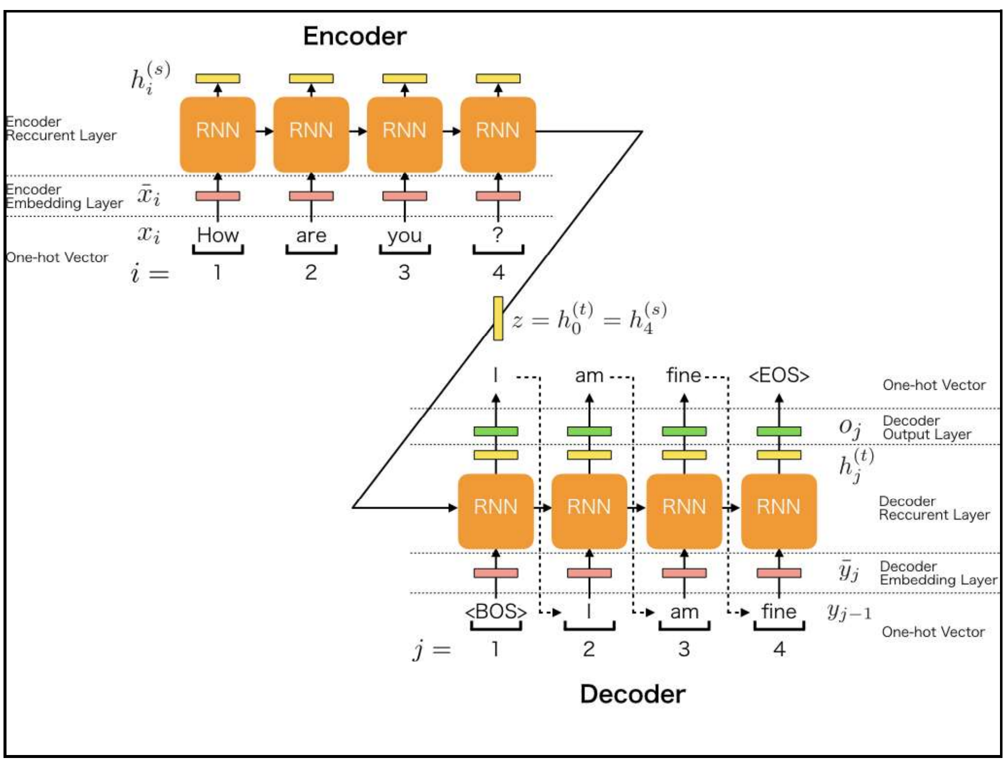


Figure 10.6: Seq2Seq model

- **Sequence-to-point**

Sequence to point learning (seq2point) operates by modifying the received network input to work as a mains window, while the output for the target appliance shows up functioning at the midpoint of its window. The model believes that the midpoint of the target appliance is meant to be correlated with the received mains signal information, both before and after the point of time when it occurs. This training technique can also be considered to be a non-linear regression.

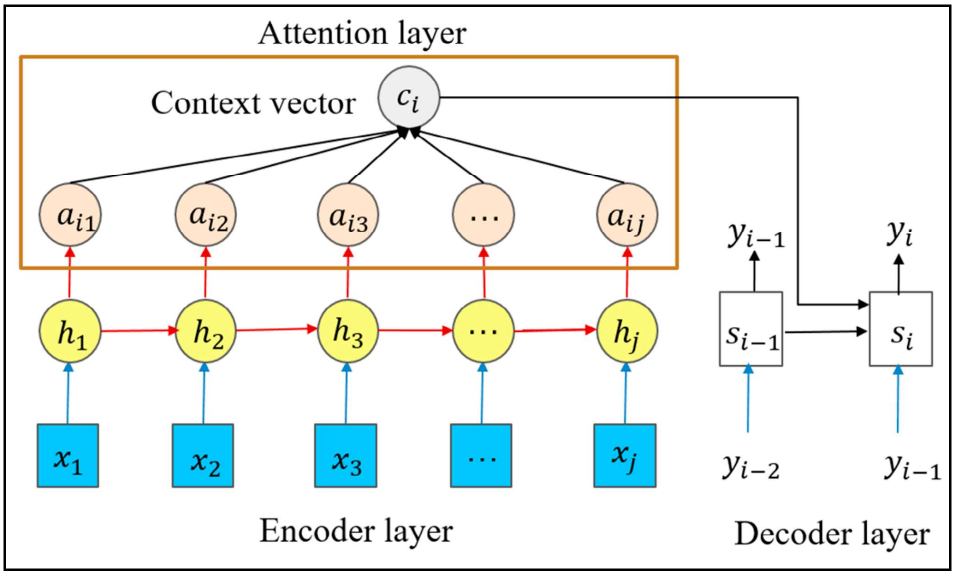


Figure 10.7: Seq2Point with attention model

- **Sequence-to-Subsequence**

Suppose Y (t) denotes the mains readings of one house at time t and Xi(t) denotes the readings of appliance i at time t. Then the total mains readings and appliance i's readings can be expressed as: Y = Y (t) T t and Xi = Xi(t) T t (i = 1, 2, ..., I), respectively, where T means the length of observed readings and I means the total number of appliances in the given house. Neural NILM,first applied deep learning technique to NILM, which learns a model from mains reading to target appliance readings with the same time window. This seq2seq approach can be formulated as Yt:t+W−1 → Xi(t:t+W−1), where W is the length of the time window. Instead of predicting a sequence of appliance readings, seq2point method try to predict the midpoint element of that window and can be formulated as Yt:t+W−1 → Xim, where m = t + b W 2 c. Although seq2point method outperforms seq2seq and yields almost the state-of-art performance, its drawback is also obvious. In seq2point approach, each forward process of the model yields only one output signal, resulting in too much computation and decreasing the utilization rate of the input, which limits the practical application of deep neural network. It would be better to choose an appropriate length of the required appliance readings according to the difficulty of the training process. Hence, we propose to train the model to predict a smaller window of appliance readings, which is called sequence-to-subsequence (seq2subseq) learning method. Without loss of generality, the center of the smaller appliance window and and the center of the mains window are aligned. That we expect to exploit the representation ability of deep neural networks and make a trade-off between the amount of computation and the difficulty level of training the model.

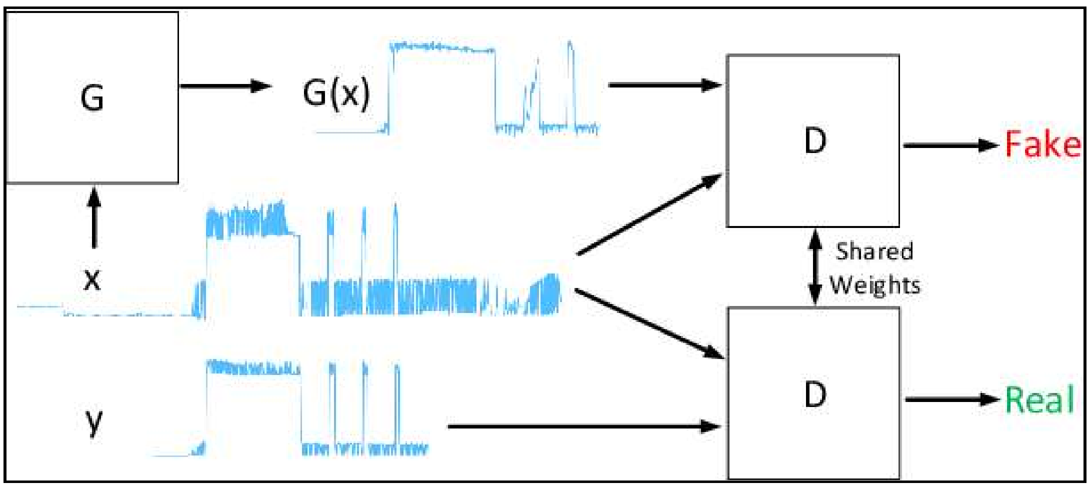


Figure 10.8: Seq2Subseq model architecture

10.3 Available Datasets

This section elaborates on the data sets that have been used for the purpose of this research experiment and the algorithms that were implemented.

- **REDD**

The Reference Energy Disaggregation Data Set (REDD), is a freely available data set containing detailed power usage information from several homes, which has been published with the aim of promoting further research on energy disaggregation. REDD was the first public energy dataset that was released by MIT in 2011. REDD contains high and low frequency energy readings from 6 households in the USA recorded for a short period (between a few weeks and a few months). This data set is widely used for the evaluation of NILM algorithms.

- **UK-DALE**

UK-DALE is an open-access data set from the UK that records Domestic Appliance-Level Electricity at a sample rate of 16kHz for the whole-house and at 1/6Hz for individual appliances. The data set contains 16 kHz current and voltage aggregate meter readings and 6 second sub-metered power data from individual appliances across 3 UK homes, as well as 1 second aggregate and 6 second sub-metered power data for 2 additional homes. An update to this data set was released in August 2015 which has expanded the data available for house 1 to 2.5 years. The updated data set has been utilized for this experiment.

- **IAWE**

The IAWE data set was released by the Indraprastha Institute of Information Technology, India. This data contains the aggregated and sub-metered electricity and gas readings from 33 household sensors, captured at a resolution of one second. The data set covers 73 days of a single house in Delhi, India.

- **REFIT**

The REFIT (`Personalized Retrofit Decision Support Tools for UK Homes using Smart Home Technology', Grant Reference EP/K002368/1/1) Electrical Load Measurements dataset includes cleaned electrical consumption data in Watts for 20 households at aggregate and appliance level, timestamped and sampled at 8 second intervals. This dataset is intended to be used for research into energy conservation and advanced energy services, ranging from non-intrusive appliance load monitoring, demand response measures, tailored energy and retrofit advice, appliance usage analysis, consumption and time-use statistics and smart home/building automation.

10.4 Electrical Device Types and Criteria of Distinction

To be able to distinguish devices within an overall load profile a NILM algorithm recognizes device specific features. These electrical fingerprints rely on the intrinsic properties of the individual devices and lead to the overall signal which can be traced by NILM. In this section we discuss the different electrical device types which can appear in a typical setup.

Ohmic Loads: The most intuitive form in NILM are ohmic loads. Consider a stove top with a nominal power of 1,500 W and an electric kettle of 2,000 W. The most obvious way to separate these two is by taking their active power intake into account. Ohmic loads have a plain active power intake and a constant impedance characteristic. This means that their power intake varies in response to fluctuating voltage. Examples for ohmic loads are the above mentioned devices, a toaster or a conventional light bulb.

Inductive or Capacitive Loads: Another type of device has coils or capacitors as internal components. An electrical motor for example uses coils to create the required magnetic field. This leads in addition to active power also to a reactive part, which can be traced within the overall signal. In addition the starting behavior of an electrical motor can be a characteristic feature. Furthermore brush sparking leads to an induced interference voltage which could be detectable by a NILM system.

Nonlinear Loads: This type of device consists of nonlinear components, like diodes. They are most commonly used to rectify AC in DC voltage and a popular example to accomplish this is the bridge rectifier. Another component which consists of nonlinear components is a phase fired control. It can vary the power supply to the corresponding load, which is for example used for a dimmable lamp. Nonlinear loads lead to a non-sinusoidal current waveform, which can be transformed into harmonics of the net frequency and be traced by a NILM system afterwards. Also magnetic saturation - for example in a magnetic core of an electric motor - can exhibit a non-sinusoidal waveform, because once the maximum magnetization is reached the sinusoidal waveform is cut-off.

SMPS, Electronic Ballasts: Switched-mode power supplies (SMPS) become more and more popular nowadays due to their efficient production size and cost. They internally work at a far higher and independent switching frequency than the net frequency. SMPS - in contrast to ohmic loads - enable a constant power supply to the corresponding load even with varying supply voltage. They emit line-conducted disturbances and therefore EMI signals in the kHz range throughout the building network and therefore can be detected by a NILM system. Electronic ballasts on the other hand limit the amount of current and are usually used to supply fluorescent lamps. Still they typically operate internally with a higher frequency of 32 and 40 kHz, which should be detectable within an EMI signal. Induction cookers are using inductively created eddy currents to heat up the cookware. They run internally on frequencies between 20 to 100 kHz and therefore should be traceable within this frequency range.

10.5 Appliances Detection NGSM Implementation

- **Using VAE on Ukdale dataset:**

Training and Testing Details:

We have trained the model on a kettle and fridge and got 90% accuracy and we can see the reconstructed signal of the load:

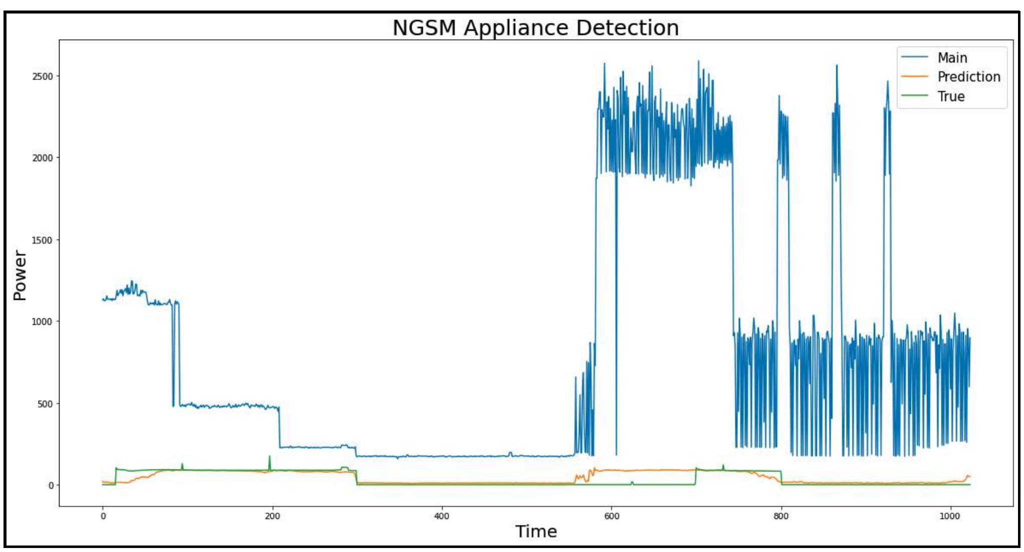


Figure 10.9: Appliance Detection Results

- **Using NGSM on edge ML model on our dataset:**

Our target is to accurately detect the working household appliances using a simple ML technique based on a random forest classifier it does a simple multiclass classification task by predicting the single working appliance or multiple of them.

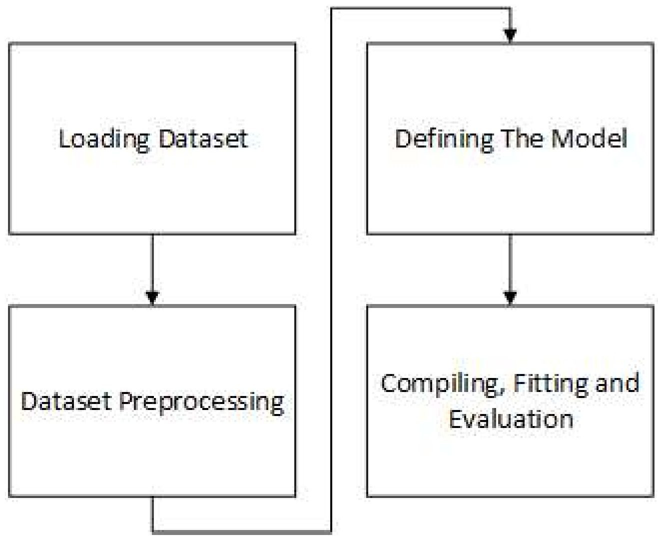


Figure 10.10: ML Model Creation Steps Flowchart

Now we will start understanding and showing the details of the implementation of the proposed approach and its steps:

- **Dataset Loading and Preprocessing**

We have collected our own dataset which contains our main 3 features Volt, Current and Power. Collected at 30 Hz from the STMP33 sensing unit.

	time	volt	current	power	label
0	1657866865	239.694	1.314	178.073	fridge
1	1657866865	239.588	1.314	180.105	fridge
2	1657866865	239.658	1.314	177.758	fridge
3	1657866865	239.588	1.317	181.178	fridge
4	1657866865	239.623	1.321	177.062	fridge
...	...	...	...	...	...
60376	1657873195	237.210	1.420	213.458	fridge_lamp
60377	1657873195	237.281	1.417	218.055	fridge_lamp
60378	1657873195	237.175	1.420	218.259	fridge_lamp
60379	1657873195	237.139	1.417	214.957	fridge_lamp
60380	1657873195	237.210	1.415	214.233	fridge_lamp

368975 rows × 5 columns

Figure 10.11: The Model Dataset

After that we splitted the data into training and testing datasets by encoding the labels with integer values.

Our final step is to Standardize or StandardScaler which standardizes a feature by subtracting the mean and then scaling to unit variance. Unit variance means dividing all the values by the standard deviation. The results we get are in a distribution with a standard deviation equal to 1 and the variance is equal to 1.

- **The Proposed Classification Model using Deep Learning**

This model is a simple random forest model to solve a simple multiclass classification problem by predicting the single working appliance or multiple of them.

- **Results**

We have collected a dataset for 3 devices and achieved 100% accuracy on the test dataset.

```
Accuracy of RandomForestClassifier classifier on training set: 1.00
Accuracy of RandomForestClassifier classifier on test set: 1.00
```

Figure 10.12: The Model Results

Deployment on Edge:

We have used the **Tensorflow library** and **learn** to make inference on edge using the Raspberry Pi zero and F1C200s.

10.6 User Recommendation System

Our system would then tell the user some recommendations in which it can reduce his overall consumption. As an example it would tell him to decrease the working hours of the heavy load appliances after a certain threshold which can be assigned by the user through the NGSM mobile application and NGSM Web-App.

10.7 Electricity Demand Forecasting

Accurate forecasts of electricity demand inform investment decisions about power generation and supporting network infrastructure. Of major interest to energy policymakers, power utilities, and private investors alike, forecasts are also essential for development professionals. Inaccurate forecasts, whether they over- or under predict demand, can have dire social and economic consequences. Underestimating demand results in supply shortages and forced power outages, with serious consequences for productivity and economic growth. Overestimating demand can lead to overinvestment in generation capacity, possible financial distress, and, ultimately, higher electricity prices.

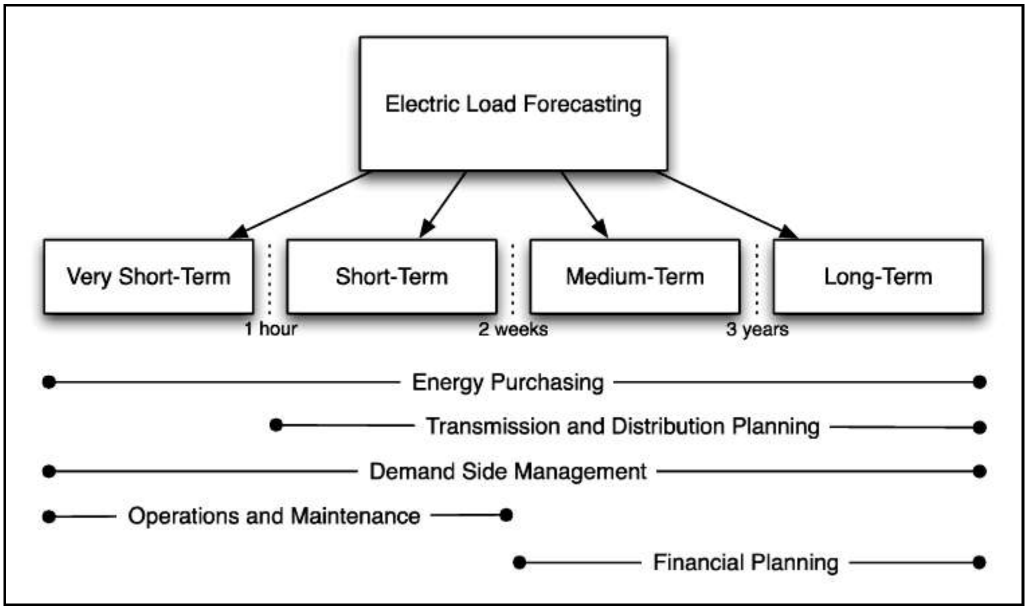


Figure 10.13: Electric load forecasting classification.

Electricity use arises from complex interactions between meteorological and socio-economic factors. Standard forecasting techniques are insufficient in such a dynamic environment, and more sophisticated methods are needed.

A good model for predicting the demand for electricity requires analyzing the following types of variables:

- Calendar data: Season, hour, bank holidays, etc.
- Weather data: Temperature, humidity, rainfall, etc.
- Company data: Price of electricity, promotions, or marketing campaigns.
- Social data: Economic and political factors that a country is experimenting with.
- Demand data: Historical consumption of electricity.

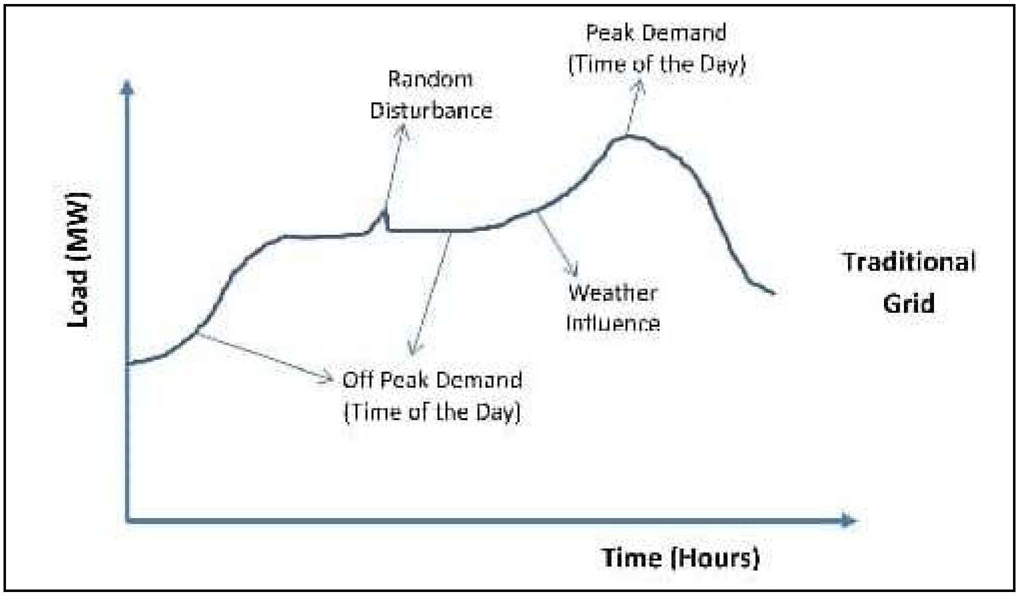


Figure 10.14: Factors that affect electrical power demand

Various types of classifications based on duration of forecasting and forecasting methods are proposed in literature over a period of time. Demand forecasting methods can be also classified in terms of their degrees of mathematical analysis used in the forecasting model. These are presented into two basic types, namely: quantitative and qualitative methods. In most cases historical data are insufficient or not available at all. Qualitative forecasting methods are generally used by planners to forecast accurately, these methods are the Delphi method, Curve fitting and technological comparisons including other methods. Other forecasting techniques such as decomposition methods, regression analysis, exponential smoothing, and the Box-Jenkins approach are quantitative methods.

Following are some of the forecasting methods which have also been used in electricity demand forecasting

- Multiple regression.
- Exponential smoothing.
- Stochastic time series.
- Fuzzy logic.
- Neural networks.
- Knowledge-based expert systems.

Some challenges facing Load forecasting can be:

1. Forecasting is based on expected conditions such as weather. Unfortunately, the weather is sometimes unpredictable and the forecasting may thus be different when the actual weather differs from expected.
2. Difficulties getting accurate data on consumption behavior due to changes in factors such as pricing and the corresponding demand based on such a price change.
3. It is sometimes difficult to accurately fit the numerous complex factors that affect demand for electricity into the forecasting models. In addition, it may not be easy to obtain an accurate demand forecast based on parameters such as change in temperature, humidity, and other factors that influence consumption.
4. The utility may suffer losses if they do not understand and decide on an acceptable margin of error in short term load forecasting.

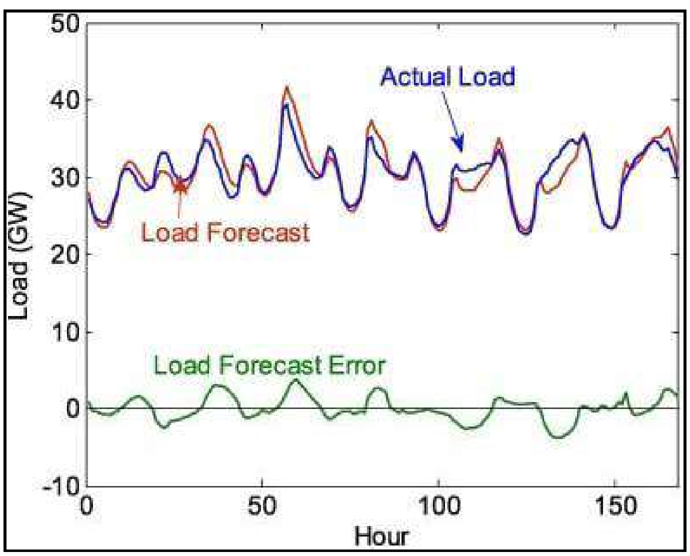


Figure 10.15: Typical load forecast error

- **Use Case**

By applying the main principles of energy demand forecasting, we used electricity data collected over the course of 5 years from houses in uk. To train a deep learning model for time series forecasting using Long Short term memory (LSTM) layers. We were able to predict the consumption of the upcoming month based on previous year data.

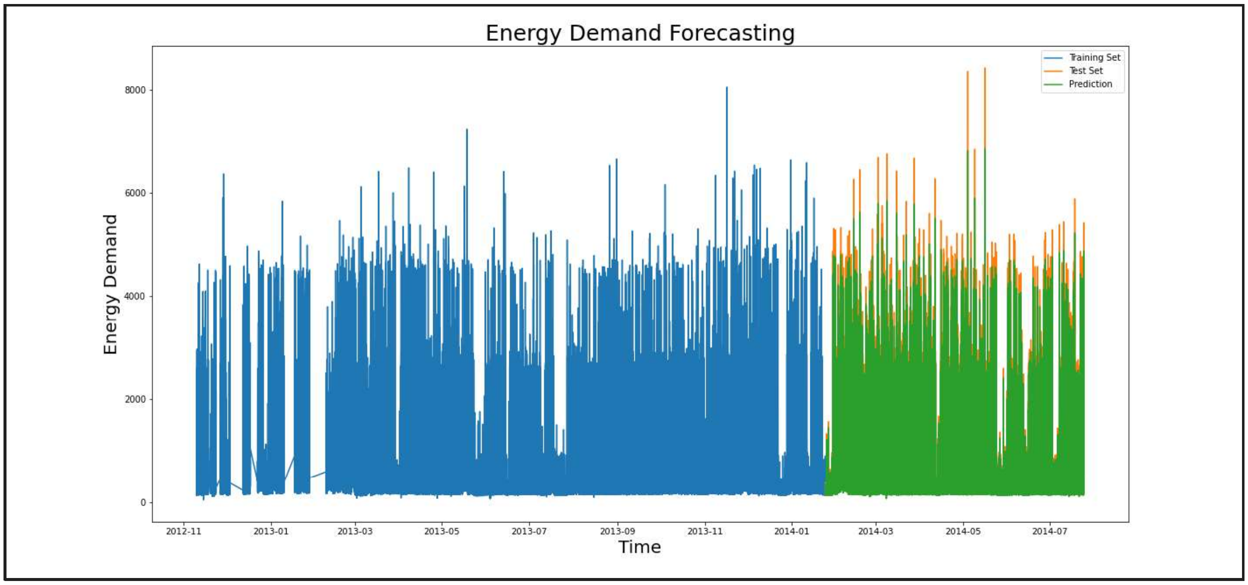


Figure 10.16: Energy Demand Forecasting Results

10.8 Warning Usage Generation

Short-Term Electricity Demand Forecasting (STEDF) provides many advantages for electricity suppliers as well as consumers. Different types of electricity demand forecasting methods are highlighted. A case study is done by selecting a medium voltage industrial consumer to illustrate the applications of STEDF. The model developed for demand forecasting can be used with smart meters to forecast the demand, calculate the maximum demand and to control the demand side loads. Furthermore, the forecasted demand can also be used to **generate warning signals** regarding maximum demand. Ultimately the proposed system will help reduce the demand and save the electricity bill for both residential and industrial consumers.

10.9 Harmonics Detection

- **Overview**

Harmonics is a distortion of the normal electrical current waveform, generally transmitted by nonlinear loads. that draw current in abrupt pulses rather than in a smooth sinusoidal manner. Switch-mode power supplies (SMPS), variable speed motors and drives, photocopiers, personal computers, laser printers, fax machines, battery chargers and UPSs are examples of nonlinear loads. Single-phase non-linear loads are prevalent in modern office buildings, while three-phase, non-linear loads are widespread in factories and industrial plants. These non-linear power supplies draw current in high-amplitude short pulses that create significant distortion in the electrical current and voltage wave shape.

- **Total Harmonic Distortion**

Harmonic distortion, measured as **total harmonic distortion (THD)**. The distortion travels back into the power source and can affect other equipment connected to the same source. THD is defined as the ratio of the power in the supply due to all the harmonics and the power of the fundamental supply.It provides an indication of the degree to which a voltage or current signal is distorted.

- **Detection Algorithm**

We used an algorithm based on fourier transform and notch filters to detect harmonics and calculate the total harmonic distortion in a given signal.

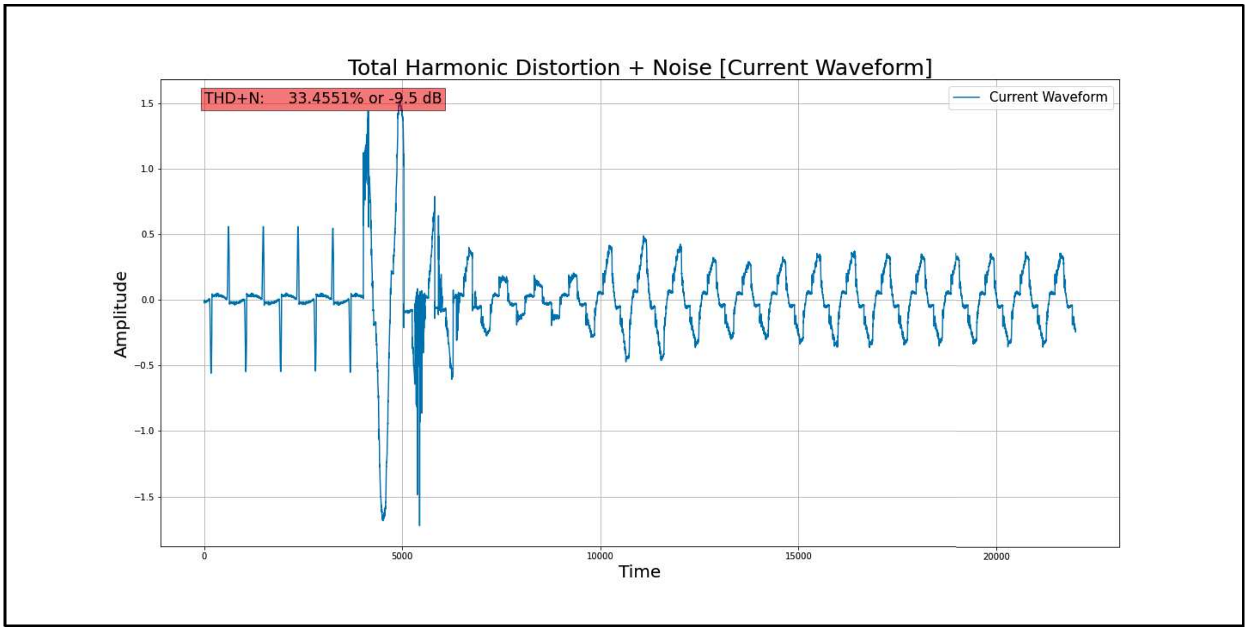


Figure 10.17: THD in current waveform

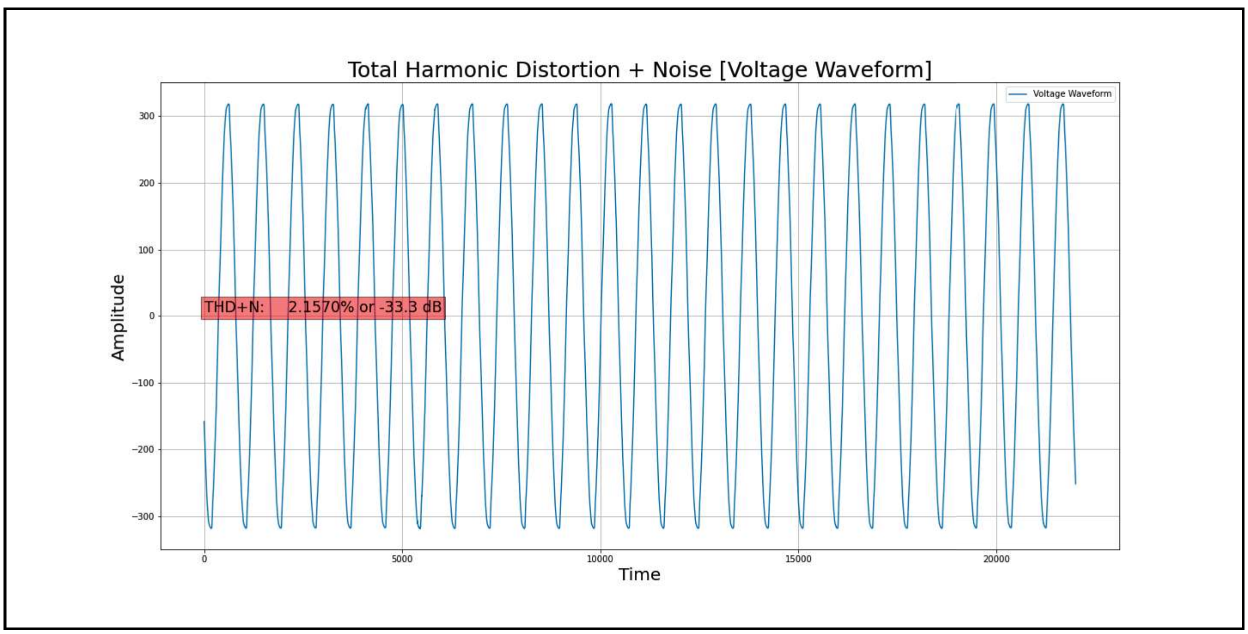


Figure 10.18: THD in voltage waveform

11. System Modeling

11.1 Use Case Diagram

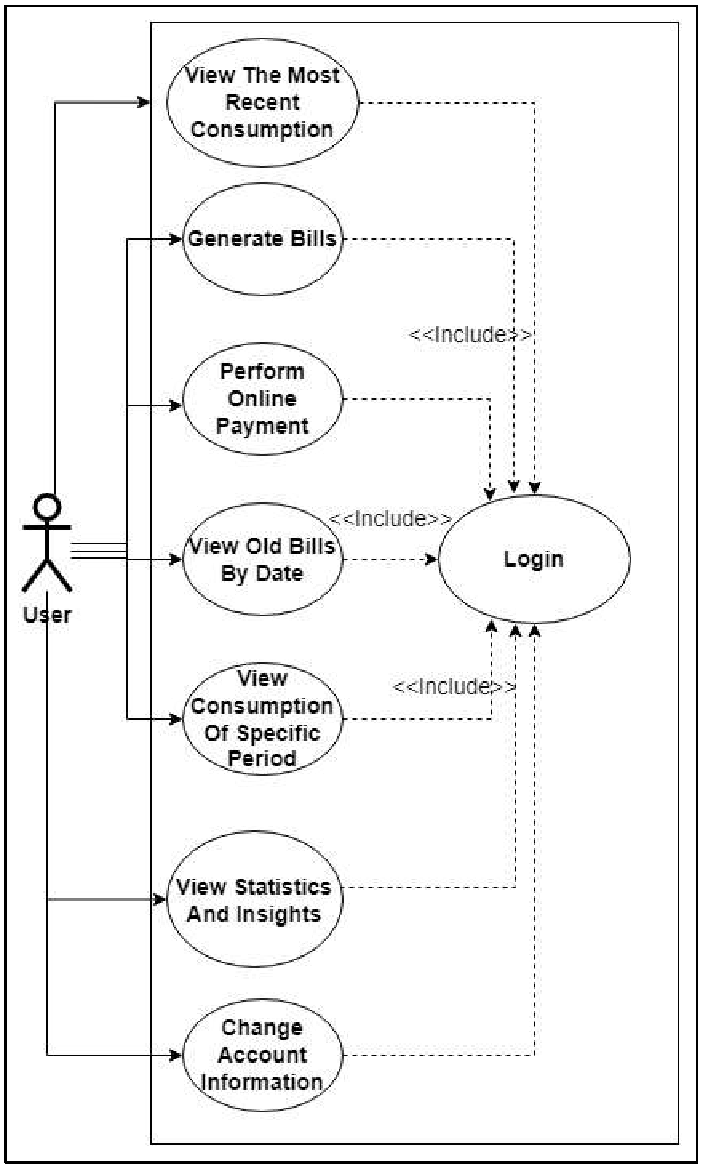


Figure 11.1: Use Case Diagram

11.2 Class Diagram

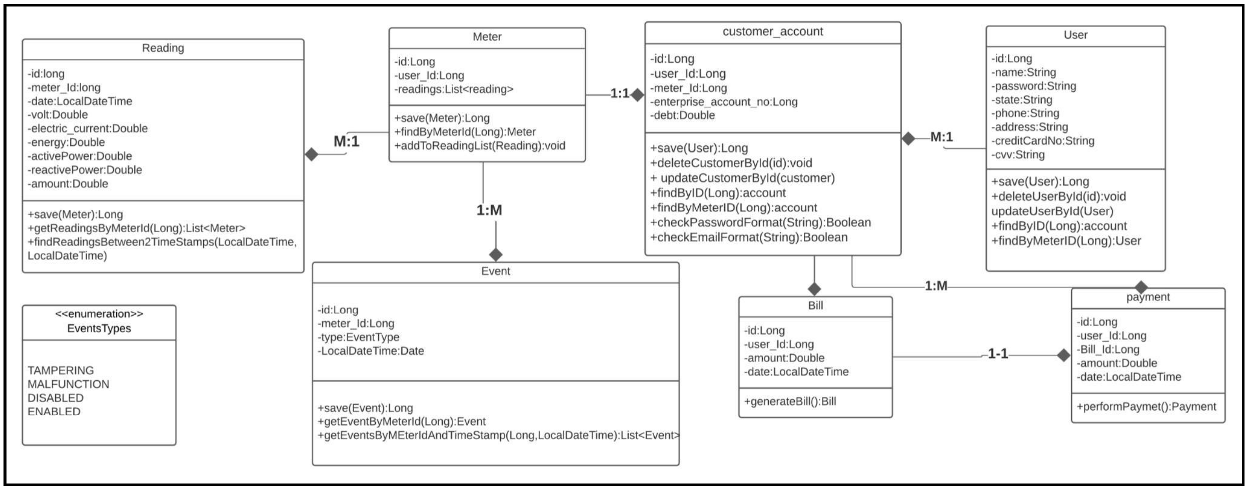


Figure 11.2: Class Diagram

11.3 Sequence Diagram

Login to the System:

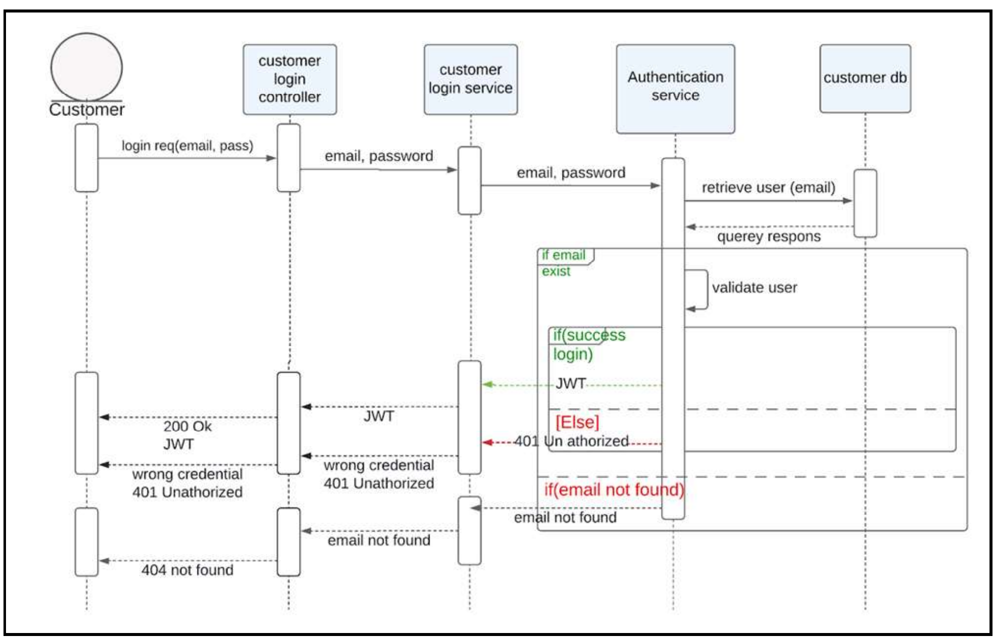


Figure 11.3: Sequence Diagram

View Consumption Sequence Diagram:

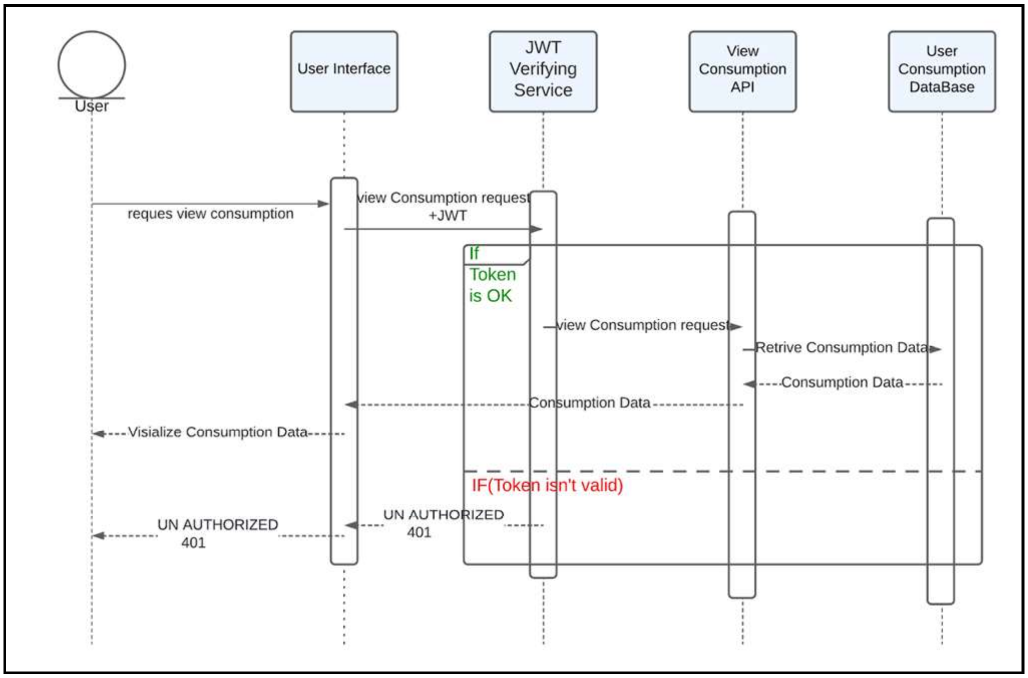


Figure 11.4: Sequence Diagram

Perform Online Payment Sequence Diagram

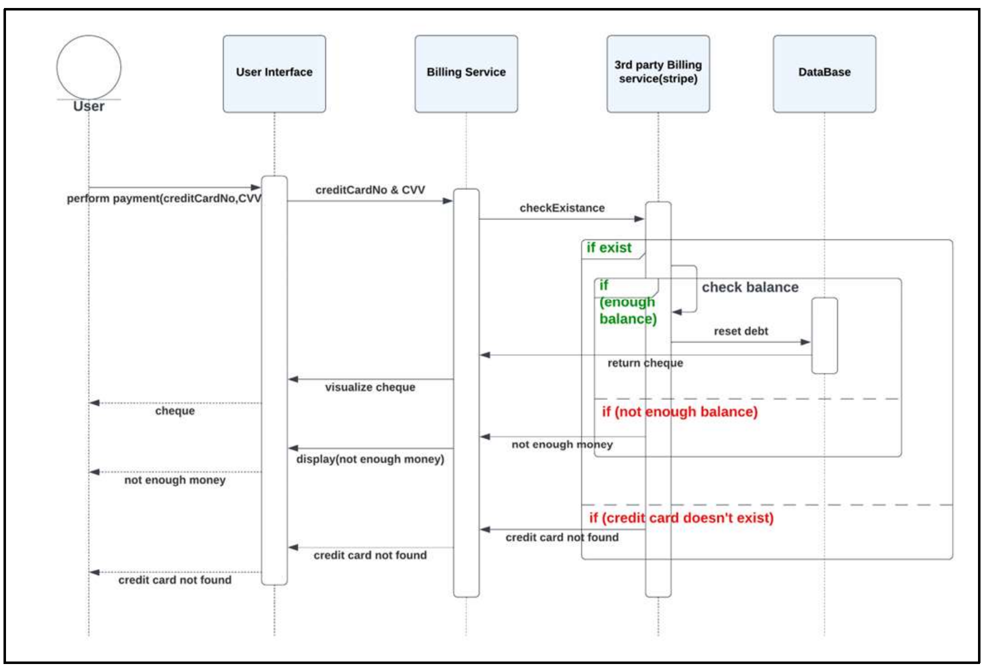


Figure 11.5: Sequence Diagram

View ML Models Results Sequence Diagram

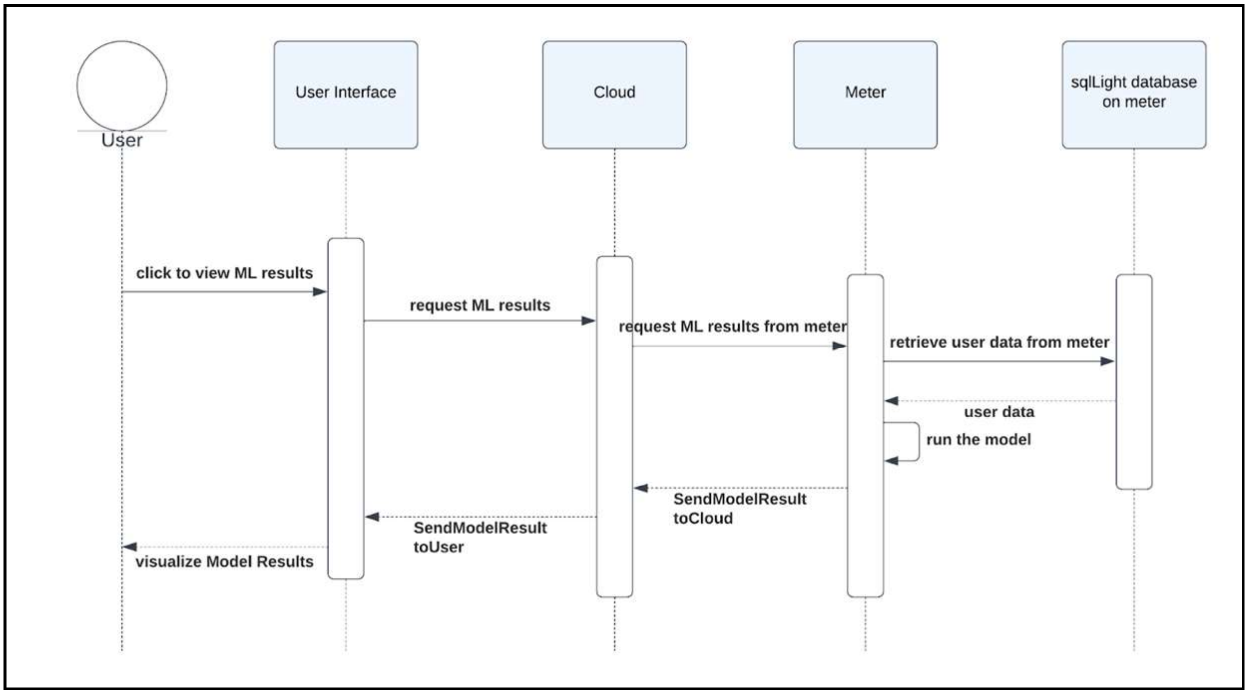


Figure 11.6: Sequence Diagram

12. Conclusion and Future Work

12.1 Conclusion:

By implementing the Advanced Metering infrastructure (AMI) we achieved more efficient monitoring on the electricity consumption which plays an important role in sustaining the electricity consumption not only that but also forecasting the future bills which helps the consumer to better understand his payment agenda. Moreover the system can detect the currently working appliances for the consumer and provide the consumer with some statistics about his usage and consumption. Our system provides payment facilities using the NGSM HES using the NGSM mobile application and the NGSM web based system. The concluded implemented NGSM system is an Embedded Linux based meter for on-edge processing with a complete Head-End System for achieving the advanced metering infrastructure (AMI). Appliance detection using the aggregated load achieved 90% accuracy with 3 devices operating on the same time. Energy demand forecasting of a collected smart meter data to predict the next 4 months from 4 years of data.

12.2 Future Work:

- Our Smart Meters are connected directly to Head End System (HES) using Peer to Peer topology, we are intending to continue the hard work to design and implement MESH Network which connects the meter with each other to the HES.
- Enhance our dataset by collecting more dataset for various appliances.
- Optimize the PCB level to be all on one PCB.
- Migrate from Monolithic to Microservice:

As it enables us to optimize resources, enhance collaboration and streamline business processes. Microservices simplify app management, making it easier for you to build, deploy, update, test and scale each service independently.

- Use Cache tier:

The cache tier is a temporary data store layer, much faster than the database. The benefits of having a separate cache tier include better system performance, ability to reduce database workloads, and the ability to scale the cache tier independently.

- Database Sharding
- Collect more appliances datasets for more classification options and electricity forecasting.

13. References

[1] *(SpringerBriefs in Applied Sciences and Technology) K.S.K Weranga, Sisil Kumarawadu, D. P. Chandima (auth.) - Smart Metering Design and Applications-Springer-Verlag Singapur (2014).*

[2] *Sayed, Sawsan & Hussain, Tasnim & Gastli, Adel & Benammar, Mohieddine. (2019). Design and realization of an open-source and modular smart meter. Energy Science & Engineering. 7. 10.1002/ese3.361.*

[3] 2022. [online] Available at: https://www.researchgate.net/publication/319162179_Electricity_Theft_Concerns_within_Advanced_Energy_Technologies.

[4] 2022. [online] Available at: https://www.researchgate.net/publication/333654345_Design_and_realization_of_an_open-source_and_modular_smart_meter.

[5] notes, e., 2022. Smart Meter Circuit Design & Block Diagram » Electronics Notes. [online] Electronics-notes.com. Available at: https://www.electronics-notes.com/articles/eco-green-engineering/smart-energy-meters/smart-meter-electronic-circuit-design.php.

[6] Iopscience.iop.org. 2022. ShieldSquare Captcha. [online] Available at: https://iopscience.iop.org/article/10.1088/1742-6596/1804/1/012207.

[7] What is a Smart Grid? - Definition from Techopedia. (2022). Retrieved 8 July 2022, from https://www.techopedia.com/definition/692/smart-grid.

[8] Smart Grid: The Smart Grid | SmartGrid.gov. (2022). Retrieved 8 July 2022, from https://www.smartgrid.gov/the_smart_grid/smart_grid.html.

[9] Smart Grids Explained. (2022). Retrieved 8 July 2022, from https://www.nanowerk.com/smart/smart-grids-explained.php

[10] Foster, I. (2022). Making Smart Grids Smarter with Machine Learning. Retrieved 8 July 2022, from

https://www.eit.edu.au/making-smart-grids-smarter-with-machine-learning/

[11] 2022. [online] Available at: <https://duepublico2.uni-due.de/servlets/MCRFileNodeServlet/duepublico_derivate_00045824/Diss_Bernard.pdf> .

https://duepublico2.uni-due.de/servlets/MCRFileNodeServlet/duepublico_derivate_00045824/Diss_Bernard.pdf

[12] 2022. [online] Available at: https://duepublico2.uni-due.de/servlets/MCRFileNodeServlet/duepublico_derivate_00045824/Diss_Bernard.pdf.

[13] Duepublico.uni-duisburg-essen.de. 2022. Lehr- und Lernmaterial in DuEPublico. [online] Available at: https://duepublico.uni-duisburg-essen.de/.

[14] Bimenyimana, S. and Asemota, G., 2022. Traditional Vs Smart Electricity Metering Systems: A Brief Overview. [online] Iiste.org. Available at: https://www.iiste.org/Journals/index.php/JMCR/article/view/42505/43773.

[15] GitHub. 2022. GitHub - torvalds/linux: Linux kernel source tree. [online] Available at: https://github.com/torvalds/linux.

[16] aliexpress.com. 2022. R3 TINY200 F1C200S DVP F1C100S| | - AliExpress. [online] Available at: https://ar.aliexpress.com/item/1005002745235950.html?gatewayAdapt=glo2ara.

[17] Data, C., 2022. Edge Computing-A cutting edge technology! - Huawei Enterprise Support Community. [online] Huawei Enterprise Support Community. Available at: https://forum.huawei.com/enterprise/en/edge-computing-a-cutting-edge-technology/thread/734895-893?page=1.

[18] What Is Machine Learning? A Definition. (2022). Retrieved 8 July 2022, from https://www.expert.ai/blog/machine-learning-definition/

[19] Deep Learning in Smart Grid Technology: A Review of Recent Advancements

and Future Prospects. (2022). Retrieved 8 July 2022, from https://ieeexplore.ieee.org/document/9395437?figureId=fig12#fig12

[20] Tiwari, A. and author, M., 2022. Electricity meters: Electromechanical meters, electronic meters and smart meters. [online] Bijli Bachao. Available at: https://www.bijlibachao.com/electricity-bill/electricity-power-electromechanical-electronic-meters-smart-reading.html.

[21] 2022. [online] Available at: https://www.eclipse.org/org/press-release/iotdevsurvey2018.php.

[22] Electric load forecasting: advantages and challenges. (2022). Retrieved 8 July 2022, from https://engineering.electrical-equipment.org/electrical-distribution/electric-load-forecasting-advantages-challenges.html

[23] (2022). Retrieved 8 July 2022, from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.866.3917&rep=rep1&type=pdf#:~:text=Different%20techniques%20namely%3B%20regression%2C%20multiple,genetic%20algorithms%2C%20fuzzy%20logic%2C%20neural

[24] GitHub. 2022. GitHub - mangopi-sbc/buildroot-mangopi-r: Buildroot Package for MPi-R1 R2 R3 R3c (F1C200s). [online] Available at: https://github.com/mangopi-sbc/buildroot-mangopi-r.

[25] Linkedin.com. 2022. Traditional meter vs Smart meter - An overview. [online] Available at: https://www.linkedin.com/pulse/traditional-meter-vs-smart-overview-gokul-shrinivas

[26] .Amazon.com. 2022. [online] Available at: https://www.amazon.com/System-Design-Interview-2nd/dp/B09559NJKL.

[27] 2022. [ebook] Available at: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://www.ijareeie.com/upload/2016/december/48_Automatic.pdf.