

# Smart Water Monitoring and Billing System Using IoT and Mobile Technology

M.P.A.M. Rathnakumara \*, A. Pallegama, R.M.T.C.B. Ekanayake

Department of Science and Technology, Faculty of Applied Sciences, Uva Wellassa University, Badulla, Sri Lanka

\* Corresponding author email address: [ayesha.r@uwu.ac.lk](mailto:ayesha.r@uwu.ac.lk)

Received: 11 September 2025

Accepted: 18 December 2025

Published: 28 February 2026

Citation: M.P.A.M. Rathnakumara, A. Pallegama, & R.M.T.C.B. Ekanayake. (2026). Smart Water Monitoring and Billing System Using IoT and Mobile Technology. Sri Lankan Journal of Applied Sciences (Online), Vol.4(2), 20–29. <https://doi.org/10.5281/zenodo.18410725>

## Abstract

This research aims to develop a smart water usage monitoring and billing system that leverages the Internet of Things (IoT) and mobile applications to replace conventional manual methods, which are often costly, time-consuming, and less reliable. The proposed system integrates a flow sensor with a Node MCU Wi-Fi module to collect real-time water consumption data and transmit it automatically to the ThingSpeak cloud server. The system calculates usage and billing at configurable time intervals, with MATLAB used for data processing and analysis. The results are delivered to a mobile app, enabling users to view their daily water consumption and charges instantly. However, this duration can be adjusted according to user requirements. MATLAB is employed to process and analyse the transmitted data, ensuring accuracy in both consumption tracking and cost estimation. The processed outputs are then delivered directly to a mobile application, providing users with convenient, continuous access to their daily water usage and associated charges. Prototype testing shows that the system reduces manual effort, lowers operational costs, and enhances transparency in billing. Overall, these findings highlight how IoT-driven innovations can significantly enhance water management efficiently while laying a solid foundation for future smart utility services.

Keywords: ThingSpeak cloud server, NodeMCU, Internet of Things

## 1. Introduction

This research is centered on real-time data transmission and updating to an online server using the NodeMCU IoT platform. As an application of this technology, an automated water metering system was designed to address inefficiencies in existing practices. In Sri Lanka, many institutions, including healthcare departments, educational systems, transport services, the National Water Supply and Drainage Board, and the Ceylon Electricity Board, still rely heavily on manual methods for data collection, processing, and analysis. These approaches require significant financial and human resources, yet they remain inefficient due to the limitations of manual labor, such as restricted working hours and vulnerability to errors. Consequently, the government incurs substantial costs, while delays and inaccuracies in data handling negatively impact overall service efficiency and the national economy. In the water sector, several additional challenges arise from the current manual system. Water meters are often installed in locations such as backyards, making access difficult for meter readers, who must sometimes wait for customers to grant entry. In cases where customers are absent, readings cannot be recorded, and the Water Board resorts to estimating consumption based on average usage. This practice, however, is

problematic from the customer's perspective, as actual consumption may exceed the estimated value, resulting in unexpectedly high bills in subsequent months due to tariff adjustments. Customers are then instructed to manually record and report their meter readings to substations for corrections, creating additional administrative burdens and costs for the billing department.

The study particularly emphasizes the issue of wastage and unbilled water, which significantly contributes to Non-Revenue Water (NRW), one of the most critical challenges faced by the Sri Lankan Water Board. High costs are also associated with water distribution, repairs, customer services, and labor, further intensifying the economic strain.

By exploring IoT-based solutions, this research aims to address these inefficiencies, reduce waste, and provide a more reliable and cost-effective approach to water management.

### 1.1 Introduction to Non-Revenue Water

In any water supply system, the total volume of water produced and released into the distribution network does not always translate into revenue for the service authority.

A significant portion is lost within the system due to leakages, inefficiencies, or unrecorded consumption. This

unrecovered volume is commonly referred to as Non-Revenue Water (NRW). [1]

It is typically expressed as a percentage of the total water supplied and is widely recognized as a critical performance indicator for water utilities. High levels of NRW pose a serious challenge to the financial sustainability of water authorities, as they directly reduce the revenue base while increasing operational and maintenance costs.

Colombo is known as the most commercialized region in Sri Lanka, as well as a very busy city. Especially the water pipelines of this city are very old built in the British era hence the pipelines run for more than 100 years up to now. Due to the complexity of this pipeline system National Water Supply and Drainage Board (NWSDB) faces so many problems especially when it comes to maintenance and repairing of these pipeline systems. With rapid urbanization, provision of safe drinking water to urban communities has become an important and challenging task for governments in developing countries.[2]

Table 1

System input volume of water [1]

System Input Volume of Water						
Unbilled Authorized Consumption	Commercial Losses	Physical Losses				
Non-Billed metered consumption	Unauthorized consumption	Metering Inaccuracies and data handling errors	Leakage on Transmission and distribution losses	Leakage on overflows at Utility Storage tanks	Leakage on service connection up	
<b>Non-Revenue water (NRW)</b>						

In Sri Lanka, water supply to urban areas, including Colombo city is provided by the NWSDB. NWSDB spends large amount of money each year to produce and distribute safe drinking water to Colombo and suburbs. NWSDB has metered all water connections and charge a fee from each water consumer, based on water volume they consume. However, a portion of water produced by the NWSDB is loss in the water supply system. This is called Non-Revenue Water (NRW) as there is no revenue generated for the water loss in the system. Many urbanized Asian cities including Colombo have high NRW levels causing large loss of revenue to water supply operators. Hence it should be considered as a major national problem. According to Table 1 NRW can take place via several means, they can be listed as,

- Physical losses
- Commercial losses
- Unbilled authorized consumption, comparatively.[3]

NRW in Sri Lanka is also at a higher rate, with the city of Colombo taking the lead. As stated by Fernando and Perera, “Currently, NRW in Colombo is about 50%, which

implies that 64 million gallons of water supplied every day to the city, 32 million gallons of water does not generate any revenue for the NWSDB”. [1,2]

Water distribution losses remain a major concern in urban areas such as Colombo, where significant volumes are recorded as Non-Revenue Water (NRW). At the same time, the water demand continues to rise both locally and globally, yet consumer awareness of daily water usage remains limited. In Sri Lanka, domestic water meters do not provide users with direct access to consumption data. Instead, billing is carried out manually every month, a process that is often prone to errors and delays. As a result, large amounts of water are lost each month through NRW, leading to considerable financial and energy wastage. To address these challenges, it is essential to adopt modern solutions that minimize inefficiencies. One such measure is the development of an advanced water metering and billing system that replaces outdated manual practices with technology-driven approaches, thereby improving accuracy, reducing losses, and promoting sustainable water management.

1.2 Problem Identification

The process of water metering in Sri Lanka is still largely manual, requiring significant financial resources from the government to maintain. Among the major challenges faced by the National Water Supply and Drainage Board is the issue of Non-Revenue Water (NRW), which represents a substantial portion of the supply lost without generating income. To mitigate this, it is necessary to establish a modernized approach to water meter reading and billing. At present, there is no real-time monitoring of water consumption at the domestic level in Sri Lanka, highlighting the urgent need for technology-driven solutions that can improve efficiency, reduce losses, and provide consumers with timely access to accurate usage information.

1.3 Objective

The primary objective of this research is to develop an IoT smart water metering system that can read water consumption, calculate usage, and generate billing amounts without requiring manual intervention. The system is designed as a real-time data transmission platform that utilizes Internet of Things (IoT) technology. Consumption data are transmitted to an online server, where they are analysed and processed, after which the results are delivered directly to a mobile application for user access.

1.4 Related Work

Efficient water governance has become a critical challenge, particularly in regions like Sri Lanka, where manual water metering and billing remain dominant. Such systems are costly, time-intensive, and prone to errors, contributing significantly to Non-Revenue Water (NRW)—water that is lost before it can be billed. Several studies have highlighted the potential of IoT platforms for water management. Fuentes, H and Mauricio, D. (2020) designed a smart water monitoring system using ultrasonic sensors and ESP8266 Wi-Fi modules, where consumption data were

transmitted to a cloud server for analysis [4]. Similarly, Miry, M., & Ibrahim, R. (2020) proposed an IoT-enabled water meter that connected flow sensors with cloud platforms, enabling real-time usage tracking.[5]

Recent research on Smart water metering has explored various IoT based solutions aimed at improving water monitoring, leakage detection, and automated meter reading. Several studies have focused on low-cost hardware-based metering.

As noted by Amir, Fauzi and Arifin (2020), smart water meter is designed using a sensor flow meter with the IoT, which offers a cost-effective solution for water resource management in smart cities. The usage of Node-MCU allows delivering real time information to the mobile application about the water flow through the sensor.[6]

According to Ray *et al.* (2020) smart water meter has low infrastructural cost, as only NodeMCU and the water flow sensor was used in recording, visualizing and analysing the water flow patterns. Excessive water flow is notified on immediate historical data and that could prevent wastage of water and ensure proper water resource management. It is a self-sustainable model as we do make use of solar panels to run the data acquisition unit and transmit the data to the Cloud platform in real time. [7]

In this paragraph it is described how, *Chnag et al.* (2020) designed a wireless transmission automatic meter reading system, established database of control centre, and the automatic water leakage detection algorithm. Through the monitoring of nodes, this system can check the data correction with software methods. The accuracy of smart sensing meter can achieve 97%, which can be further promoted to near 100% after the software correction. By this way, this system can monitor the water flow and accurately search the location of water leakage. The detection time for one section is about 1ms, which can achieve real-time monitor. [8]

The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals, or people that are equipped with unique identifiers (UIDs) and the capability to transfer data over a network without requiring human-to-human or human-to-computer interaction.[9]

IoT devices integrated with cloud services provide an efficient alternative to manual monitoring. Cloud platforms such as ThingSpeak have become popular for handling IoT-generated data. For instance, Ganesan *et al.* (2024) implemented a water quality and consumption monitoring system where flow and turbidity sensors transmitted data to ThingSpeak for visualization. [10]

ThingSpeak is an IoT platform that is designed to enable meaningful connections between people and things. Features of ThingSpeak is real-time data collection, data analysis, data processing of the position information, data visualization, and message transmission using connected SNS, via an open-source API to support various platforms. It helps easy transfer of data from embedded devices such as Arduino, NodeMCU, and Raspberry Pi. It supports various languages and environments such as C, Node.js, Python and visual programming languages. ThingSpeak's MATLAB

analytics toolbox further allows automated calculations. This integration of cloud services reduces the need for complex local storage and processing. [3]

The NodeMCU IoT device is an ESP8266 is a chip that manufacturers use to create wirelessly networkable microcontroller modules. ESP8266EX is among the most integrated Wi-Fi chips in the industry. Measuring of 5mm\*5mm, ESP8266EX requires minimal external circuitry a 32-bit Ten silica MCU, standard digital peripheral interfaces, antenna switches, RF module, power amplifier, low noise receive amplifier, low noise receive amplifier, low noise receive amplifier, filters and power management modules all in one small package. [11]

Even though IoT-based smart water metering has advanced significantly especially in areas like low-cost hardware design, leak detection, and cloud-based data visualization the majority of current research only looks at specific aspects of the water management process. Previous research usually concentrates on automated meter reading, cloud storage, leak detection, or real-time monitoring, but it does not combine these features into a cohesive, end-to-end solution. Additionally, very few systems are created with automated invoicing, domestic user accessibility, or the practical deployment requirements of poor nations, where human meter reading still predominates and causes errors, delays, inefficiencies.

As a result, there is a need to create an all-encompassing IoT-based framework that seamlessly integrates measurement, cloud analytics, automatic billing, and mobile access into a single, user-friendly solution appropriate for practical home applications. This study covers the mentioned research gap by integrating all above mentioned categories.

### 1.5 Expected Outcomes

The purpose of this research is to propose an IoT smart water meter reading and billing system that can be accessed directly by users through a mobile application. Consumers are provided with an Android-based app that allows them to log in and view both their current water usage and billing information. The system employs a digital water meter connected to a NodeMCU IoT platform, which transmits readings to the ThingSpeak cloud server. Within the server, dedicated channels are used to receive data from IoT sensors and represent them graphically in real time. Processed results are then forwarded to the mobile application at regular intervals of 10 minutes, though this can be adjusted as required. MATLAB is used to analyse the transmitted data and perform calculations related to consumption and billing. By offering instant updates, the system increases user awareness of daily water usage while ensuring accuracy and reliability. Compared to traditional manual methods, the proposed solution is cost-effective, requires minimal human intervention, and is capable of continuous operation without the limitations of human labour, making it a practical approach to modernizing water management.

## 2. Materials and Methodology

### 2.1 Materials

Water flow sensor – Module YF S401, NodeMCU board – Model ESP8266MOD, Male jumper wires, LED bulbs, Rubber pipeline.

### 2.2 Methodology

The water flow sensor was interfaced with the NodeMCU Wi-Fi module, where the black wire was connected to the ground (GND), the red wire to the 3.3V supply, and the yellow wire to the D2 pin. The sensor operates on the principle of a rotor mechanism, generating pulses proportional to the flow rate as water passes through. These pulses were processed by the NodeMCU to calculate real-time water consumption.

The flow sensor measures the amount of water flow through the pipes. Pulse frequency is directly proportional with flow rate of the sensor.

Liquid Flow Rate (Q) = Average Velocity of flow (V)\* Cross Sectional area of pipe (A);

$$Q = V * A;$$

Pulse Frequency (Hz)/ 4.5 = Q (flow rate in Liters per minute); -----Equation 1

Liters = Q\*t (time taken in seconds)/ (60) (seconds/minutes); -----Equation 2

From Eq.1 & Eq.2:-

Liters = (Pulse frequency/4.5)\*t/60; -----Equation 3

Pulse Frequency = Pulses/Second; -----Equation 4

From Eq.3 & Eq.4 :-

Liters = [(Pulses/seconds)/4.5]\* (t) time taken in seconds /60(second/minute);

Liters = Pulses/ (4.5\*60); -----Equation 5

The amount of water flow through flow sensor can be measured by Equation 5.

Here 4.5 is the calibration factor. It varies with the diameter of the flow sensor.[6]

The settings should be adjusted in the Arduino software before programming NodeMCU.

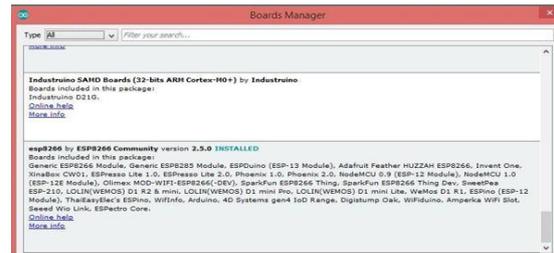
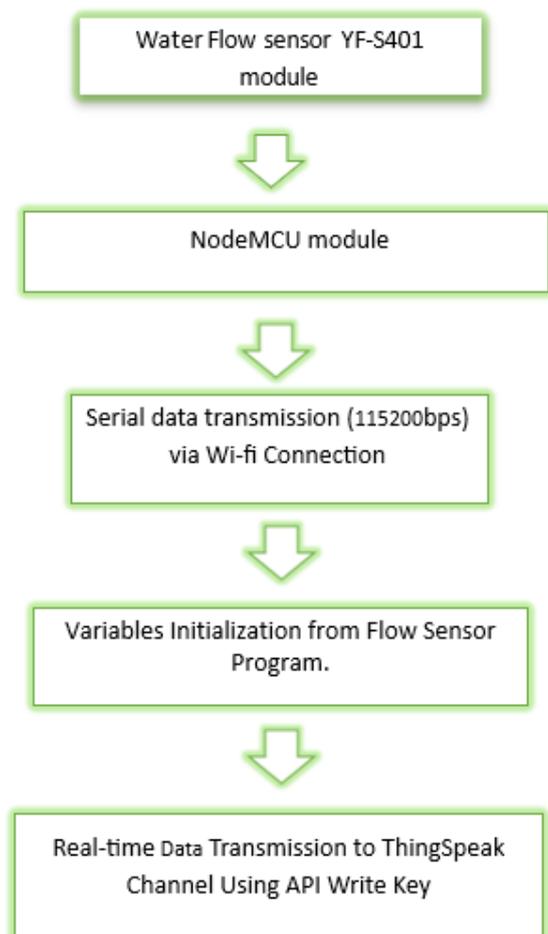


Fig.1. Esp2866 version 2.5.0 installation

The module was programmed using the Arduino IDE, with the ESP8266 NodeMCU library (version 2.5.0) installed. The board configuration was set to *NodeMCU 1.0 (ESP-12E Module)*, communication port COM4, and an upload speed of 115200 bps. Figure 01 depicts NodeMCU library Esp2866 version 2.5.0 was installed from the internet.



Block diagram 01: IoT smart water system.

In this study, an IoT-based smart water metering system was developed using a NodeMCU module connected to a YF-S401 ultrasonic flow sensor. The finalized program was uploaded to the NodeMCU to enable continuous measurement and transmission of water flow data through

Wi-Fi to a dedicated ThingSpeak cloud channel. All necessary hardware pins were assigned and calibrated to ensure accurate pulse detection from the flow sensor, and an interrupt-based counting method was used to capture real-time variations in water flow. The ThingSpeak platform was prepared by creating a channel named 'Water Usage' with four fields to display flow rate, fractional values, flow in millilitres, and total cumulative flow. The system was configured to connect to Wi-Fi and transmit the captured data to the cloud using the Channel's unique API Write Key, allowing secure uploading of measurements into the corresponding fields. Once configured, the NodeMCU transmitted live readings to the cloud, where the real-time data sent to the ThingSpeak server where it was automatically stored, processed, and visualized as graphs for monitoring and analysis.

The overall process of the system is illustrated in the block diagram 01 provided for methodology, which outlines the flow from sensing, data processing and Wi-Fi transmission to cloud storage and visualization.

The four fields are mentioned here.

Field 01: (field1) displays the flow rate.

Field 02: (field2) displays the 'frac' amount. Here frac means,

'frac' = (flow rate - int (flow rate)) \* 10; flow rate is declared as float function. It contains decimal numbers. Int function only gives integer values. Therefore, frac determines the amount of fractional part as a multiplier of 10.

Field 03: displays the 'flowMilliLitres'. It indicates the current flow in millilitres within one second.

Field 04: Displays the 'totalMilliLitres'. It displays the total flow through flow sensor. This calculates the cumulative millilitres amount.

Before uploading the code, ThingSpeak channel settings should be adjusted.

The channel settings of the ThingSpeak online server are shown in Figure 02. Each channel is assigned a unique ID and, in this study, the channel was named "Water Usage." Four fields were configured: 'Flow Rate', 'Frac', 'Flow', and 'TotalFlow'. After the real-time data were transmitted from the NodeMCU module, they were automatically displayed and visualized as graphs within the channel.

The screenshot shows the 'Water Usage' channel settings page. At the top, it displays the channel ID (891332), author (ayesham97), and access (Public). Below this are tabs for Private View, Public View, Channel Settings (selected), Sharing, API Keys, and Data Import / Export. The 'Channel Settings' section shows a progress bar for 'Percentage complete' at 30%. The channel ID is 891332. The name is 'Water Usage'. There is a description field. Four fields are configured: Field 1 (FlowRate), Field 2 (Frac), Field 3 (Flow(MilliLitres)), and Field 4 (TotalFlow(MilliLitres)). A 'Help' section on the right provides instructions for channel settings, including how to use the 'Percentage complete' field, how to enter a unique name, how to enter a description, how to check the 'Fields' box, how to enter metadata, how to enter tags, and how to link to an external site.

Fig. 2. ThingSpeak channel settings

By using real-time data in the channel, the billing amount and water consumption can be found. To find water consumption within a specific time period, MATLAB is used. As shown in the Figure 03, MATLAB analysis was done to calculate the water consumption within 10 minutes. Time can be adjusted as the consumer wants.

The screenshot shows the 'Water Usage' channel page. At the top, it displays the channel ID (891332), author (ayesham97), and access (Public). Below this are tabs for Private View, Public View, Channel Settings, Sharing, API Keys, and Data Import / Export. There are buttons for 'Add Visualizations', 'Add Widgets', and 'Export recent data'. The 'Channel Stats' section shows the channel was created 2 months ago, last entry 23 days ago, and has 1426 entries. There are buttons for 'MATLAB Analysis' and 'MATLAB Visualization'.

Fig. 3. ThingSpeak Online channel

The MATLAB analysis option in ThingSpeak was utilized to calculate water consumption. Real-time data were updated to the online channel at 15-second intervals. The *ReadAPIKey* was employed to access the channel, with flow and total flow values retrieved specifically from Field 3 and Field 4, respectively.

For an example to retrieve the last 3 data from field 4 in given channel,

```
Data = thinkspeakread (channelID, 'Fields', [4], 'NumPoints', 3, 'readkey', readAPIkey),
```

```

Apps / MATLAB Analysis / Water Usage And Billing / Edit
Name
Water Usage And Billing
MATLAB Code
1 %calculate total amount of water usage and finding the billing amount for 10 minutes periodica
2 % Channel ID to read data from
3 readChannelID = 891932;
4 % total flow(ml)
5 totalFlowFieldID = 4;
6 %flow in every 15 seconds
7 flowFieldID =3;
8 % Channel Read API Key
9 % My channel is private, entering the read API Key between the '' below:
10 readAPIKey = '532XKPQW16EMVBC';
11 % Get sum of the flow data for the last 10 minutes from the thingspeak channel field 3 and
12 % get last entry data from field 4(total flow).
13
14
15 a = thingspeakRead(readChannelID,'Fields',[totalFlowFieldID],'NumPoints',40,'ReadKey',readAPIK
16 b = thingspeakRead(readChannelID,'Fields',[totalFlowFieldID],'NumPoints',1,'ReadKey',readAPIKe
17 c = thingspeakRead(readChannelID,'Fields',[totalFlowFieldID],'NumPoints',39,'ReadKey',readAPIK
18
19 % Calculate the total amount of water usage in 10 minutes.
20 x = sum(a)/1000;
21 z = sum(c)/1000;
22 Q = x-z;
23 % Get last entry value of total flow and turn it into litres.
24 y = b/1000;
25 %Total amount of water usage in litres within 10 minutes.
26 sumwater = (y-Q);
27

```

Fig. 4. MATLAB code

The last 40 data points from Field 4 (Total Flow) in the channel were retrieved and stored as variable ‘a’, representing the real-time data transmitted over the previous 10 minutes. Data were sent to the channel at 15-second intervals, meaning one minute corresponds to 4 data points and ten minutes to 40 data points. The most recent data point from Field 4 was also retrieved separately and assigned to variable ‘b’.

The last 39 data points from Field 4 (Total Flow) in the channel were retrieved and stored as variable ‘c’.

The sum of the last 40 data points was calculated, and the result was converted to Liters by dividing by 1000.

$$x = \text{sum}(a)/1000;$$

Then, the sum of the last 39 data points was calculated.

And answer was taken in litres by dividing by 1000.

$$z = \text{sum}(c)/1000;$$

$$Q = x-z;$$

Here, Q represents the total flow recorded for the 10-minute period prior to the most recent data.

Then the last total flow data uploaded to the channel was also converted into Liters.

$$y = b/1000;$$

The water consumption over the last 10 minutes was calculated by subtracting the total flow recorded 10 minutes earlier (Q) from the most recent total flow reading. This calculated value was assigned to the variable ‘sumwater’.

$$\text{sumwater} = y-Q;$$

Total bill can be calculated from the water consumption.

The values of Cost per Liter and Service Bill can be adjusted as required. Table 2 presents only assumed values, since the calculation was performed based on 10-minute water consumption. However, these assumptions can be replaced with actual tariff rates used by the National Water Supply and Drainage Board if the calculation is extended to one month.

Table 2

Cost of Water Consumption and Service Charge

No. of Litres used (sumwater)	Cost per Litre (Rs)	Service charge (Rs)	Total bill (Rs)
0-1	0.2	-	Sumwater*0.2
1-4	0.4	-	Sumwater*0.4
4-8	.06	5	(Sumwater*0.6)+5
8-12	0.8	10	(Sumwater*0.8)+10
>=12	2	25	(Sumwater*2)+25

Structured programming in MATLAB was employed to calculate the total bill as shown in the Figure 05. Decision-making was implemented using conditional statements, where Boolean expressions were evaluated as true or false. Specifically, *if-elseif-else* statements were used to handle different billing conditions.

```

41
42 % Calculate the bill for the water usage amount.
43 if sumwater ==0
44     Totalbill =sumwater*0 ;
45 elseif (sumwater > 0 ) && (sumwater <= 1)
46     Totalbill =sumwater*0.2 ;
47 elseif (sumwater > 1 ) && (sumwater <= 4)
48     Totalbill =sumwater*0.4 ;
49 elseif (sumwater > 4 ) && (sumwater <= 8)
50     Totalbill =(sumwater*0.6)+5 ;
51 elseif (sumwater > 8 ) && (sumwater <= 12)
52     Totalbill =(sumwater*0.8)+10 ;
53 else Totalbill =(sumwater*2)+25 ;
54
55 end
56 % Display the total amount of water usage in 10 minutes.
57 display(sumwater,'water consumption within 10 minutes(Litres):');
58 % Display bill amount for water usage in 10 minutes.
59 % water cost in Rupees.
60 display(Totalbill,'Total bill amount within 10 minutes(Rupees):');
61
62 % Replace the [] with channel ID to write data to:
63 writeChannelID = [888407];
64 % Enter the write API Key between the '' below:
65 writeAPIKey = '0U12Y08FHR067K';
66 % Replace total water consumption and bill to another channel within 10 minutes.
67 thingspeakWrite([888407,'Fields',[1,2],'values',{sumwater,Totalbill},'writeKey','0U12Y08FHR066
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```

Fig. 5. MATLAB based Analysis for Bill Calculation

The total bill and water consumption values were updated in a separate ThingSpeak online channel using output commands.

By utilizing the ‘TimeControl’ feature of the ThingSpeak online server, the MATLAB code was scheduled to run automatically at 10-minute intervals. Consequently, the system recalculated and updated the water consumption and corresponding bill every 10 minutes.

Once the water consumption and total bill were calculated automatically, the information was transmitted to the Android application. Users could access their consumption details and billing information directly through the app. The mobile application was developed using MIT App Inventor, where the interface was designed by arranging necessary elements and components. The initial screen included a user authentication feature with Sign Up/Login functionality, as shown in the Figure 07.



Fig. 6. Android app design

Here visual programming environment was used to code the app. Block-based coding programs were set to sign up or log in system.

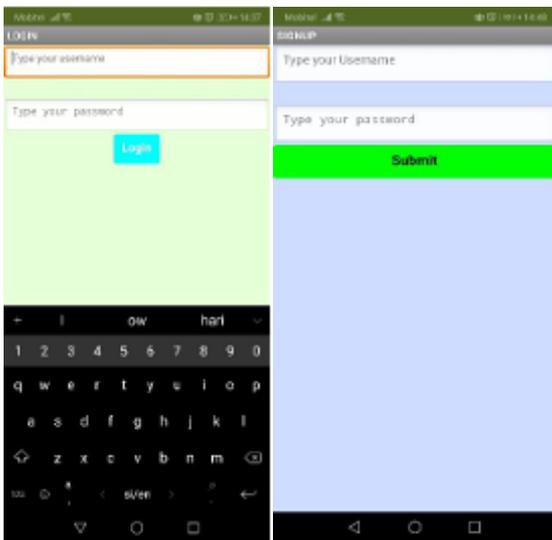


Fig. 7. Sign Up/Login functionality

When the username and password was entered to sign up process, both words were saved in database in MIT App Inventor. The database used in the application is TinyDB, which provides a method for storing values locally within the app. The stored data persists on the phone even after the

application is restarted, and each app is associated with only a single data store. When the user enters a username and password in the designated fields, these values are stored in TinyDB. Upon successful sign-up, the login screen is displayed. When the username and password are entered on the login screen and the submit button is pressed, the application navigates to the home page.



Fig. 8. Design of the Home Page

When the ‘Retrieve Water Consumption and Bill’ button was pressed, the water usage and bill values were displayed on the home page in two separate fields. The application logic for this functionality was implemented using MIT App Inventor’s block-based programming interface as shown in Figure 09. The mobile application was connected to the ThingSpeak online server through a URL. This URL, along with the channel’s read API key, enabled the app to retrieve the most recent data uploaded to the server. Variable initialization was also performed during this process.

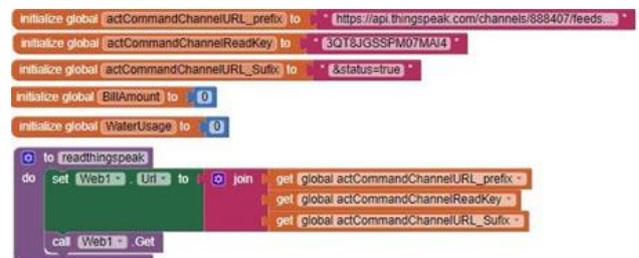


Fig. 9. Code snippet connecting the app to the ThingSpeak server.

When the ‘Retrieve Water Usage and Bill Amount’ button was clicked, the application displayed the water consumption in Field 1 and the bill amount in Field 2. These values were updated automatically every 10 minutes with the

latest water consumption and billing data. Once the Android application was fully designed, it was compiled and downloaded onto a mobile device for use.

### 3. Results and Discussion

When water passes through the flow sensor, pulses are generated corresponding to the flow. The NodeMCU Wi-Fi module, connected to the sensor, counts the total number of pulses. Using the pulse count and a calibration factor, both the flow rate and total flow are calculated. The results are then displayed on the serial monitor. Figure 10 illustrates the output of the water flow sensor.

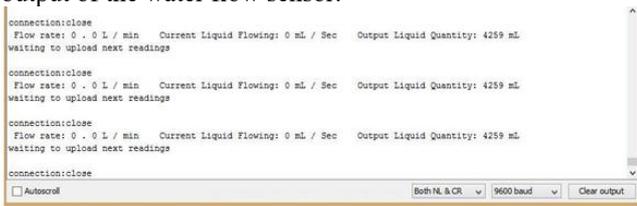


Fig. 10. Output of the Water Flow Sensor

The values updated to the online server can be graphically represented as in Figure 11.

MATLAB analysis was performed using the collected data to calculate water consumption and billing at 10-minute intervals, incorporating both cost per litre and service charges.

For example: 4.26 Liters used is Figure 12. According to the Table 02, this value is between 4-8 group, the cost of a Liter is Rs 0.6 and the service charge is Rs 5,

$$(4.26 * 0.6) + 5 = 7.56$$

therefore, the Total bill is Rs. 7.56.

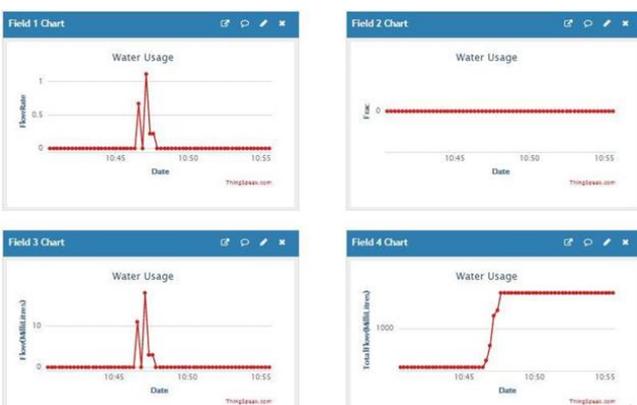


Fig. 11. ThingSpeak channel outputs



Fig. 12. Water consumption and Billing cost

As shown in the Figure 13, 0.66 Liters of water were used.

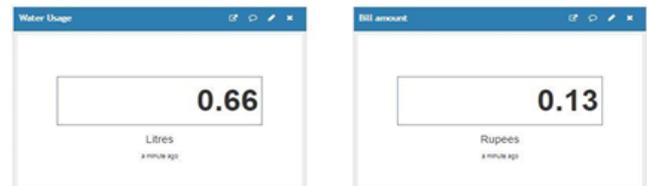


Fig. 13. Water consumption and Billing cost

This value is in between 0-1 group in Table 2. Therefore, cost of a Liter is Rs 0.2, and the service charge is Rs 0,

$$(0.66 * 0.2) = 0.13$$

Total bill is Rs. 0.13.

The cost per Liter and service charges used here are assumption values, not real rates. Actual values can be obtained from the National Water Supply and Drainage Board and substituted accordingly. This system can be applied in domestic settings, eliminating the need for manual meter readings. The outputs from the ThingSpeak online server channel were then transmitted to the Android application.

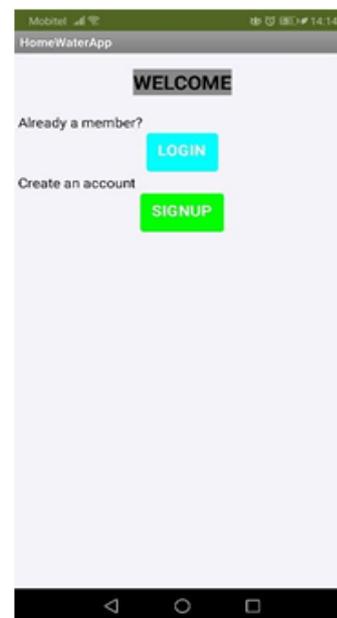


Fig. 14. User Interface of Android Application

The Water Usage and Billing app was accessed on a mobile phone, where the username and password were entered as depicted in Figure 14. After pressing the 'Retrieve Water Usage and Bill' button on the home page, the information was displayed in the Android application.

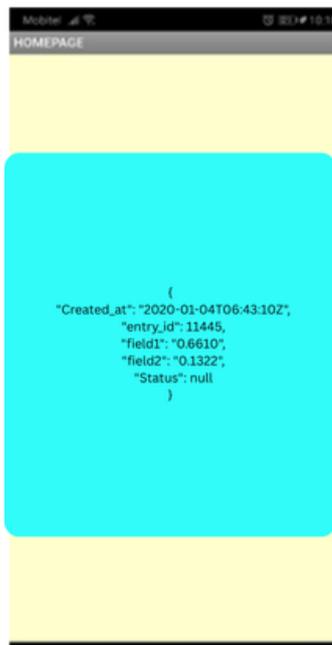


Fig. 15. Water Usage and Billing – User Interface

Field 1 shows the water consumption in Liters, while Field 2 displays the bill amount in rupees. The entry ID represents the last time the MATLAB code was executed using the channel data.

#### 4. Conclusion

The proposed system demonstrates that an affordable, Scalable and fully connected smart water management solution is not only feasible but also highly effective for domestic applications. By integrating low-cost hardware with cloud computing and mobile technology. This approach modernizes the traditional metering practices and empowers consumers with actionable insights into their water usage. The work presented here therefore, contributes a practical, end to end framework that can support broader digital transformation initiatives in the water sector and help automated and user accessible metering.

#### 5. Future Directions

In future, several enhancements can be introduced to strengthen the current design. The features are improving the accuracy of flow measurement with higher-precision sensors, adding backup power options such as a solar

module. The mobile application can be uploaded with additional features such as leakage alerts, usage predictions and personalised consumer reports. These improvements create more stable and advanced foundation, allowing the system to be effectively scaled and integrated into wider smart city infrastructures.

#### Conflicts of Interest

There are no conflicts to declare.

#### Acknowledgments

My foremost gratitude is expressed to my supervisor Dr. R.M.T.C.B. Ekanayaka, Senior Lecturer, Department of Science and Technology, Faculty of Applied Sciences, Uva Wellassa University of Sri Lanka for his constant encouragement, kind advice and generous supervision provided in the completion of this project. I acknowledge Mr.K.M.U.Dias, Electrical Engineer, National Water Supply and Drainage Board, Monaragala, for the guidance, advice and support to make the project a success. I, myself, wish to acknowledge R.U.A.U. Pallegama and all the academic staff and technical officers of the Department of Science and Technology, Uva Wellassa University of Sri Lanka. I convey special thanks to my parents and friends who encourage me in numerous ways for the success of the project. Finally, I would like to express my sincere gratitude to all the people whose names have not appeared but whose untiring effort was very much crucial to the success of this study throughout this work.

#### References

- [1] Malithi W.P.P., 2016 Proceedings of 7th International Conference on Sustainable Built Environment paper id: ICSBE2016-75
- [2] Fernando,W.B.G. and Perera,T.W.S. (2010) The Issues and Challenges of Reducing Non-Revenue Water, Sri Lanka.
- [3] Jang, R., Jung, S.J. and Soh, W.,2015 Proceedings of International Conference on Chemical, Material and Food Engineering. Atlantis Press.on Design and Implementation of Data Report Service for IoT Data Analysis.
- [4] Fuentes, H., Mauricio, D. Smart water consumption measurement system for houses using IoT and cloud computing. *Environ Monit Assess* 192, 602 (2020).]
- [5] Miry, A.H. and Aramice, G.A. (2020) 'Water monitoring and analytic based thingspeak', *International Journal of Electrical and Computer Engineering (IJECE)*, 10(4), p. 3588.
- [6] Amir, A., Fauzi, R. and Arifin, Y. (2022). Smart water meter for automatic meter reading. *IOP Conference*

Series: *Materials Science and Engineering*, 1212(1), p.012042. doi: <https://doi.org/10.1088/1757-899x/1212/1/012042>.

- [7] Ray, A. and Goswami, S. (2020). IoT and Cloud Computing based Smart Water Metering System. *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*. doi: <https://doi.org/10.1109/parc49193.2020.236616>.
- [8] Hsia, S.C., Wang, S.-H. and Hsu, S.-W. (2020). Smart Water-Meter Wireless Transmission System for Smart Cities. *IEEE Consumer Electronics Magazine*, pp.1–1. doi: <https://doi.org/10.1109/mce.2020.3043997>.
- [9] Polianytsia, A., Starkova, O. and Herasymenko, K.,2017, Proceedings of 4th International Scientific-Practical Conference Problems of Information Communications. Science and Technology on Survey of the IoT data transmission protocols.
- [10] Ganesan P., Hameed T. and Maruthakutti M., (2024) 'IOT based water flow meter using ThingSpeak', *International Journal of Science and Research Archive*, 11(2), pp. 1956–1962.
- [11] Rasadurai, K., Kalanidhi, K. and Vinodkumar, D.,2017, Smart City Billing for Homes through IoT. *International Journal of Engineering Research in Computer Science and Engineering* 4, 73-79