INTRAVENOUS DRIP MONITORING SYSTEM USING HCSR-04

J Component Report in

MICROPROCESSOR AND INTERFACING

CSE 2006

submitted by

TEAM MEMBERS:

Team Members	Reg. No.
Shaunak Deshmukh	20BCE0043
Biswajeet Raut	20BCE2465
Sandeep Raj	20BCE0053
Bidwan Rath	20BCE0681

in partial fulfilment for the award of the degree of

B.Tech

in

COMPUTER SCIENCE ENGINEERING

Under the guidance of

Faculty: Dr. Ajitha D

School of Computer Science and Engineering

NOVEMBER 2022

TABLE OF CONTENTS

ACKNOWLEDGEMENT

I would like to thank our professor **Dr. AJITHA D** for giving us this opportunity to perform this project and guiding us about how to build and develop the project and ideology. I'm are extremely thankful for the keen interest he took in advising us, for the books and reference materials provided for the moral support extended to us.

Also, I am very grateful to Vellore Institute of Technology for this wonderful opportunity. It was a great leaning experience and I plan to take our idea further and develop it as real- world project.

Shaunak Deshmukh: 20BCE0043
Biswajeet Raut: 20BCE2465
Sandeep Raj: 20BCE0053
Bidwan Rath: 20BCE0681

ABSTRACT

During recent years, we have seen much technological advancement which help to take better care of patient's health and assure them fast and safe recovery. The most basic item necessary is competent patient care in hospitals, as well as proper management of fluid and electrolytes. In hospital almost all patients, mainly ICU patients, have to regulate the volume of fluids and electrolyte into the bloodstream 24X7 which is done using drip. These drips need regular monitoring or changing to maintain constant flow of fluids or to prevent any infection to patients. Though this system is simple, overcrowded hospitals and the risk of deficiency of nurses can put patients at health risk which can deteriorate patients' health more aggressively or in worse case it can lead to death. Almost in all hospitals, nurses or the hospital staff are responsible for monitoring the drip level. But unfortunately, because of their hectic schedule, the observer may forget to change the bottle at the appropriate time. During the pandemic, hospitals were overrun with patients, and nurses were unable to do manual regular checks on the drip conditions and drip level of every patient, even after working extra shifts. Many patients even died due to not being able to get proper care from nurses. The next step in providing more effective and easy healthcare is to automate such vital procedures. To overcome this critical situation, we are proposing a NodeMCU base Intravenous Drip Monitoring System using HCSR-04 which eases the process of measuring and solves the issues of bubble formation in drips.

INTRODUCTION

Intravenous therapy is a process of administering medicine into the body directly through the veins. It is the fastest way of delivering fluids and medications throughout the body as it utilizes the effective cardiovascular channel and its natural pumping forces. The working principle of IV is that a bottle filled with the desired fluid medication is hung at a level higher than the patient's body to provide the fluid with a pressure, generated by gravitational potential energy, to overcome the cardiovascular pressure.

IV therapy is an easy and effective procedure which allows a drop-by-drop administration of medication. The process also has some innate limitations, like formation of tissue in the needle, rolling of patient on the tube or over the hand, can block the flow of the fluid and compromises with the patient's life in severe scenarios.

The major bottleneck of the process is monitoring the medicine bottle. This is necessary so that the doctor or the nurse can know when to change the bottle. In general, the bottle is changed when it is completely empty. The time it takes to empty is variable and depends on parameters such as medicine quantity used, changes in back-pressure due to systolic-diastolic and non-quantifiable dilation or contractions of veins.

Because of the dynamic nature of this process, measurement based robust monitoring is necessary. Failure of this can lead to various medical complications like backflow of blood into the IV setup. In severe cases, if the bottle gets empty and it is not monitored for some time, it can also cause insertion of air embolism in the IV tube, which can be deadly and hence critical monitoring is required. Currently, this monitoring is done by the nurses and/or by the patient's attendant and in countries like India, they have a poor nurse to patient ratio.

This project deals with design, fabrication and testing of an Intravenous Drip Monitoring System using HCSR-04, which is capable of precisely monitoring drip level in the saline bottles for the patients, and giving an alert message to the concerned doctor or nurse when the level falls below a specific reading as decided upon by the concerned doctor or nurse.

OBJECTIVES

The main objective of our system is to develop the Intravenous Drip Monitoring system by using Node MCU.

This project organizes-

- To achieve a low cost and safety healthcare in Intravenous Drip System.
- To monitor the fluid level by the HCSR-04 Ultrasonic Sensor.
- To ensure safety of the patients by this IV Monitoring system.
- To provide ease of accessibility in changing the saline bottles for the nurse.
- To ensure high degree of reliability and low cost in making this IV Monitoring System.

RELATED WORK AND LITERATURE SURVEY

01. Nicola Giaquinto, Marco S carpetta and Mattia Alessandro Ragolia proposed "Real-time drip infusion monitoring through a computer vision system" in 2020 IEEE Explore. The method proposed in this system monitors intravenous infusion using Microwave Time Domain Reflectometry. A method based on deep learning computer vision for IV drip monitoring was proposed. This technique was found to be less invasive than other available solutions, which require a direct contact with the infusion kit, offering good accuracy performance anyway. The advantage of this system is that the the control system can be better in time consumption and the system can easily control the hardware.

02. Shiyong Zheng, Zhao Li and Biqing Li proposed "The design of liquid drip speed monitoring device system" in AIP Conference proceedings – 1864, 020123 in 2019. The main function of this design is to address the unattended patients in the process of having an intravenous drip, nonadjustable solution velocity and the insufficient treatment after having an intravenous drip. This design mainly adopts infrared photoelectric sensor. The SCM AT89C51 chip is a small and complex system, which includes hardware design and software design in this whole system. The advantage of this system is that the smallest system of SCM is small in size, simple in structure and is good in repeatability and high in security based on MCU.

03. M.Anand, M.Pradeep, S.Manoj, L.Marcel Arockia Raj and P.Thamaraikani proposed the "Intra Drip Monitoring System" in IIJSR(Indo Iranian Journal of Scientific Research) January – March 2021. The methodology used is that the IV fluid monitoring system automatically sends a message to the nurse through GSM technology and automatically turned off the flow of a liquid from the IV system by using the solenoid valves. The advantage of this system is that the control system can be the better in time consumption and the system can easily control the hardware.

04. Sumalatha Bandari, Gauri Deshmukh, Pooja Pawar, Rutuja Yadav, Komal Jagadale and Dipti Chavan proposed “IV Drip Monitoring and Control System” in International Journal for Research in Applied Science and Engineering Technology(IJRASET) in 2022. The methodology proposed in this system is that a load sensor is used to determine the status of liquid in the bottle whether it is normal or warning status. The output obtained from the sensor is then processed to check whether the saline bottle is empty or not and displayed on the LCD display. When the level of saline drops below a certain level; the application will notify the user. The bottle level sensor with load cell is used to measure the saline level.

05. Satwik Tanwar, Deepa. K, Riddhi Maniktalia and Ritvik Billa proposed “IoT Based Drip Infusion Monitoring System” in International Journal of Science and Research (IJSR) in 2020. This project proposes a method for hospitals to efficiently monitor drip infusion levels. Our proposed system consists of an Ultrasonic sensor to detect the level of fluid and a light sensor to detect any bubble formation in the IV infusion bag. A control mechanism can alert the nurses or doctors if the fluid levels in the IV infusion bag drops down a certain level to prevent air embolism and avoid reverse flow of blood. It also prevents any chance of human error caused due to overflow in the hospital. This idea not only provides an automatic & reliable way to monitor the fluid percentage in patients IV drip but also to detect air bubble formation which causes risk to patient’s health.

06. Nikhil Kumar Patidar and Ravi Yadav proposed “Development of Real-Time Monitoring & Controlling of Intravenous Drip Infusions Using the Internet of Things (IoT)” in the Journal of Emerging Technologies and Innovative Research (JETIR) in 2021. The microcontroller of the Arduino family ESP-32 has been used to control and manage the system; the power supply of the whole model is DC. Once the power is ‘ON’, a welcome message is displayed on LCD, then the Arduino Board ESP-32 searches for the programmed SSID. After that, it connects with the Wi-Fi module embedded in the board with asking for password and trying to connect with Wi-Fi router. Once Wi-Fi is successfully connected, microcontroller starts capturing the liquid level of IV INFUSIONS with the help of contactless sensors at three levels (Top, Middle, Bottom) of the IV drip and send it to cloud data collector ‘Thing Speak.

07. Sanjay and Sanju Vikasini. R proposed “IOT BASED DRIPS MONITORING AT HOSPITALS” International Journal of Science and Research (IJSR) in 2020. This system works based on Arduino AT Mega 328, GSM module sim 800a, 1kg load cell, HX711 Amplifier. A load cell is be fixed on the saline stand with the load cell amplifier HX711. The saline bottle will be hanged at the center of the load cell so that it can measure the weight of the bottle. Arduino Uno will be in the patient’s room and once the bottle has reached to 40 percent of saline, the Arduino will be activated and it glows the LED bulb (red color to indicate emergency) which will be placed in front of the patient’s room. By this way, anyone passing the room will be able to identify that it is an emergency and intimate the nurses nearby. Later when the saline level reaches to 20 percent, Arduino will alert GSM Sim 800a used to send a message to mobile phones of all the nurses in duty indicating the need to change the saline bottle. This will hence give time for the nurses to replace the bottle accordingly.

HARDWARE COMPONENTS USED

- **NodeMCU DEVKIT 1.0:**

The NodeMCU Development Kit is based on ESP8266 and it integrates GPIO, PWM, IIC, 1- Wire and ADC all in one board. It has Arduino like hardware IO, Advanced API for hardware IO, which can dramatically reduce the redundant work for configuring and manipulating hardware. It has Nodejs style network API and Event-driven API for network applications, which facilitates developers writing code running on a 5mm*5mm sized MCU in Nodejs style. It also has FCC certified Wi-Fi module and PCB antenna.

- **HCSR-04 Ultrasonic Sensor:**

It is an ultrasonic sensor, also known as an ultrasonic transducer that is based on a transmitter and receiver and mainly used to determine the distance from the target object. The amount of time it takes to send and receive waves will determine how far the object is placed from the sensor. It mainly depends on the sound waves working on “non-contact” technology. The required distance of the target object is measured without any damage, giving you accurate and precise details. This sensor comes with a range between 2cm to 400cm and is used in a wide range of applications including speed and direction measurement, wireless charging, humidifiers, medical ultrasonography, sonar, burglar alarms, and non-destructive testing.

- **Esp8266-01 Wi-Fi Module:**

The ESP8266 ESP-01 is a serial to Wi-Fi breakout module with a built in ARM microprocessor that has 1MB of memory and 2 GPIOs brought out to the header for connecting to peripherals. It can be used as a serial to Wi-Fi bridge to add Wi-Fi capability to a project or it can even be programmed directly and used as a little stand-alone processor. It has full TCP/IP capability built-in. It uses a 32-bit RISC Tensilica Xtensa LX Processor running at 80MHz and a Flash Memory of 1MB and a 3.3V operation.

SOFTWARE COMPONENTS USED

Firebase:

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in Realtime to every connected client. When we build cross-platform apps with our Apple platforms, Android, and JavaScript SDKs, all of our clients share one Realtime Database instance and automatically receive updates with the newest data. The Firebase Realtime Database can be accessed directly from a mobile device or web browser; there's no need for an application server. Security and data validation are available through the Firebase Realtime Database Security Rules, expression-based rules that are executed when data is read or written. Firebase apps remain responsive even when offline because the Firebase Realtime Database SDK persists your data to disk. Once connectivity is re-established, the client device receives any changes it missed, synchronizing it with the current server state.

VSCode:

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages and runtimes (such as C++, C#, Java, Python, PHP, Go, .NET). It is made by Microsoft with the Electron Framework, for Windows, Linux and macOS.

In the Stack Overflow 2021 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool among 82,000 respondents, with 70% reporting that they use it. It provides comprehensive code editing, navigation, and understanding support along with lightweight debugging, a rich extensibility model, and lightweight integration with existing tools.

➢ **Arduino IDE:**

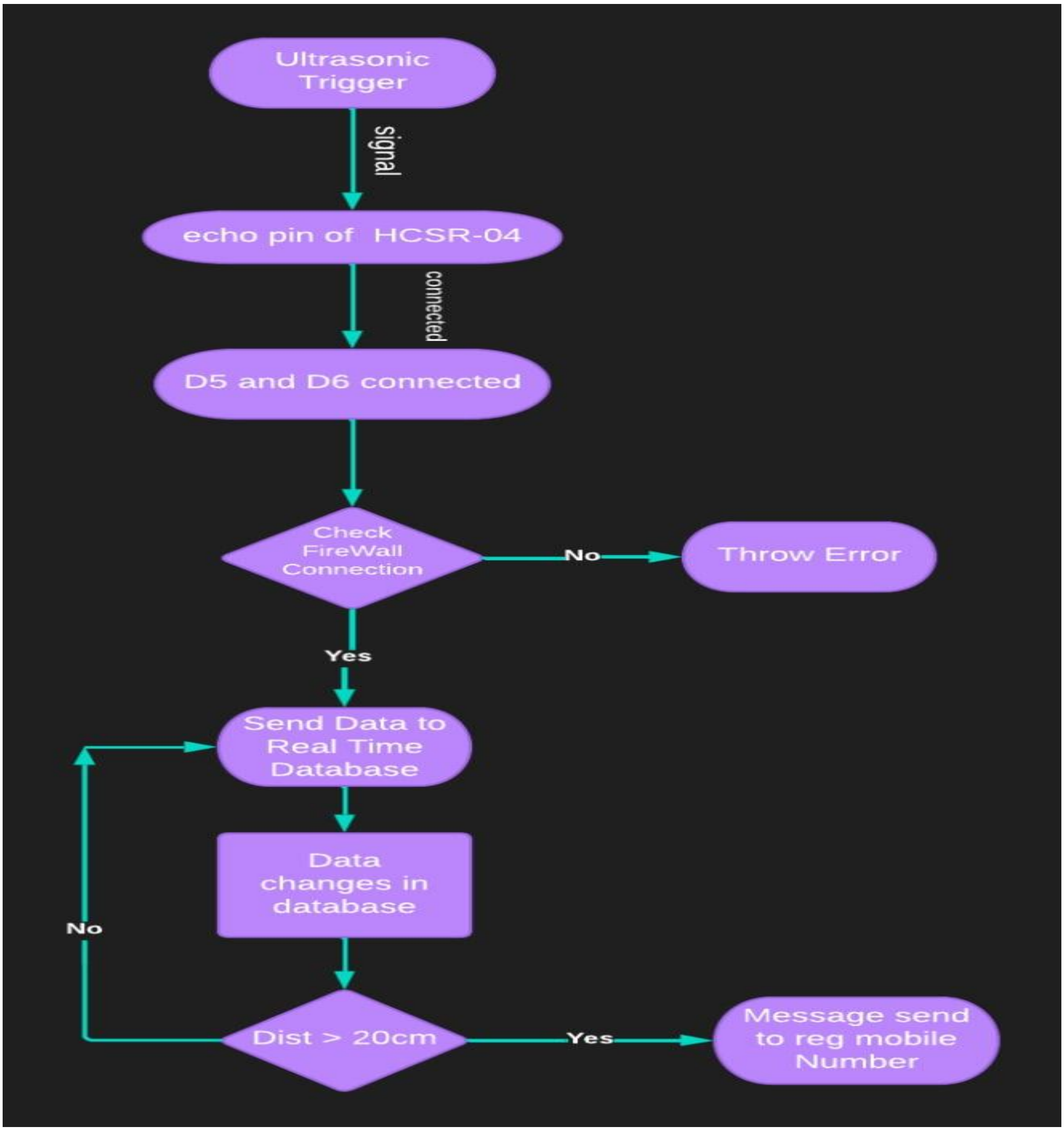
The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

Programs written are called sketches. These sketches are written in the text editor and are saved with the file extension ".ino". The Arduino Software (IDE) uses the concept of a sketchbook: a standard place to store your programs (or sketches). Beginning with version 1.0, files are saved with a ".ino" file extension. Previous versions use the ".pde" extension.

PROPOSED METHODOLOGY

The proposed methodology is to provide help to hospital staff to keep a check on the drip level of the patient. So, we use NodeMCU and Ultrasonic sensor to detect and send the distance of the drip level from the sensor level. After the detection we connect the data with the Realtime database service provided by firebase (Google Cloud Services). This is data is constantly updated as the drip level changes. Now this data is being collected or fetched by the website and it shows the percentage of saline remaining and it sends a message to the hospital staff's email id.

FLOWCHART:



WORKING

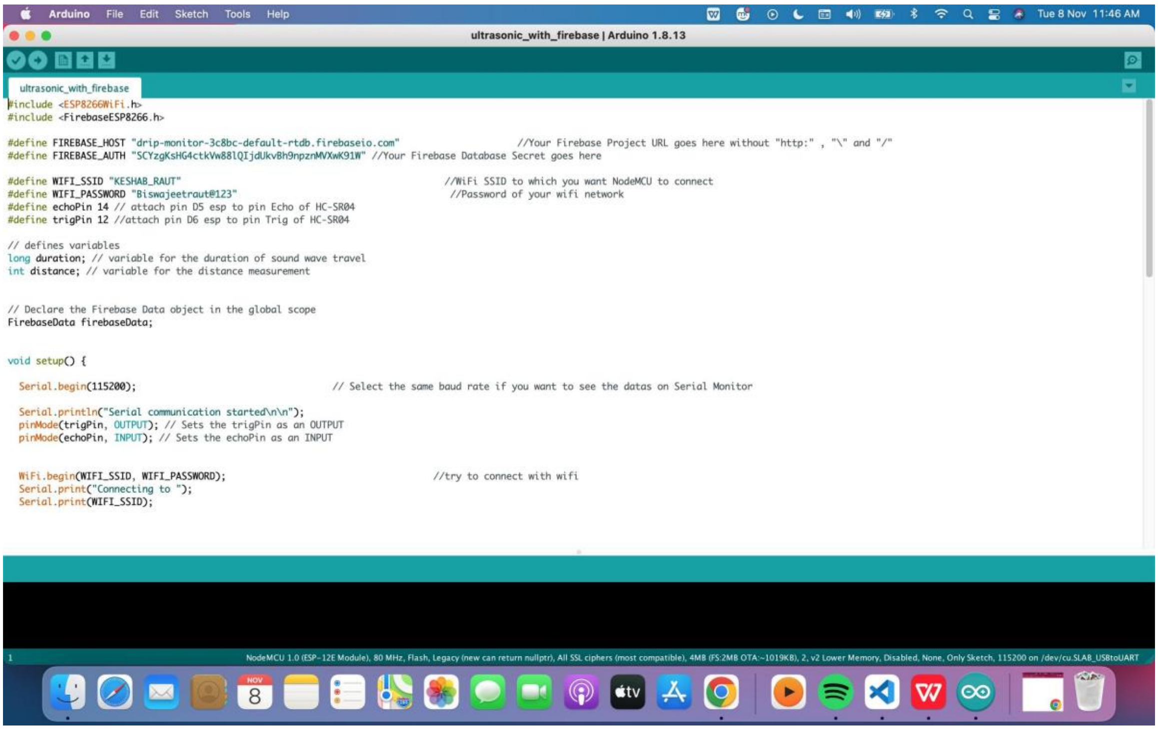
First the trigger pin triggers the ultrasonic sound pulses and then the sound pulses get reflected back and these are received by the receiver part of the Ultrasonic sensor. When the pulses are reflected back to the sensor then the echo pin produces a pulse.

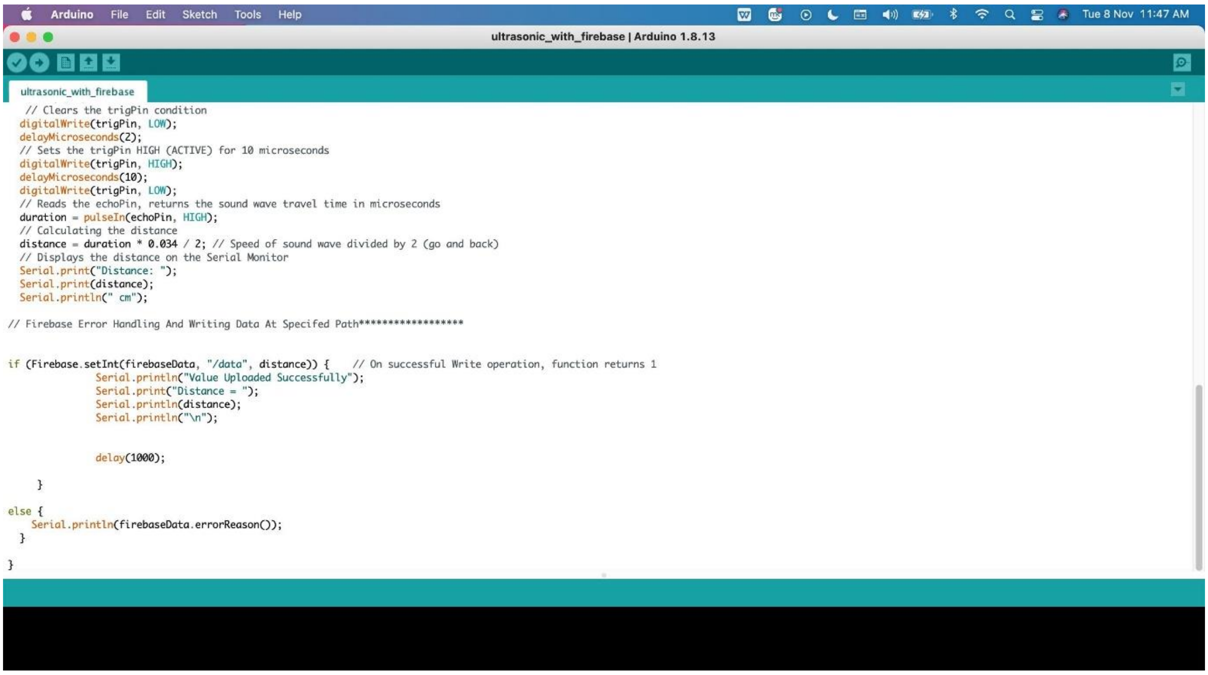
These are connected with the NodeMCU pins. The data is transferred to the D5 and D6 pin of NodeMCU. Then, by the Arduino code, we establish a connection with the firebase Realtime database and the ultrasonic sensor. The data is now updated in the database continuously.

In the website, the hospital staff has to login through their given credentials. The monitoring starts and the website fetches data of the level in cm. Now, the data is being checked against the standard value by the hospital staff and if the level is less than 20 percent, then a mail goes to the hospital staff each time the level changes. This process will continue till the staff presses the stop button.

IMPLEMENTATION AND DESIGN

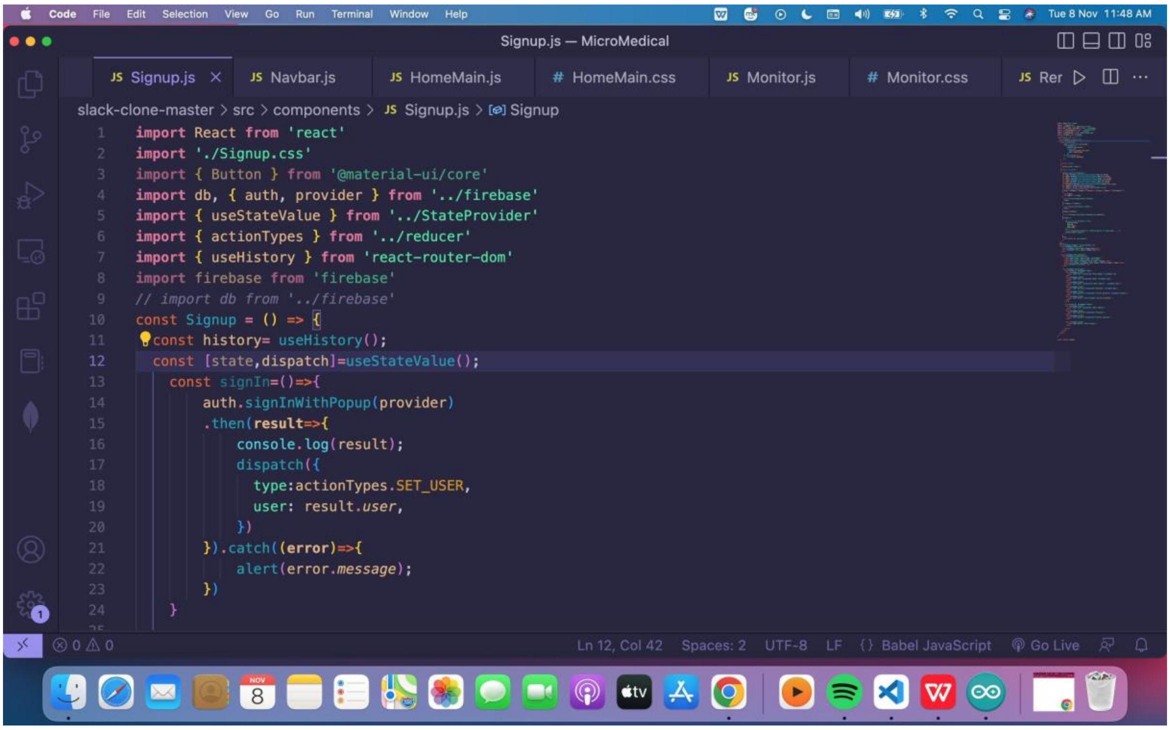
- **Arduino code for Ultrasonic Sensor with Firebase Connection:**



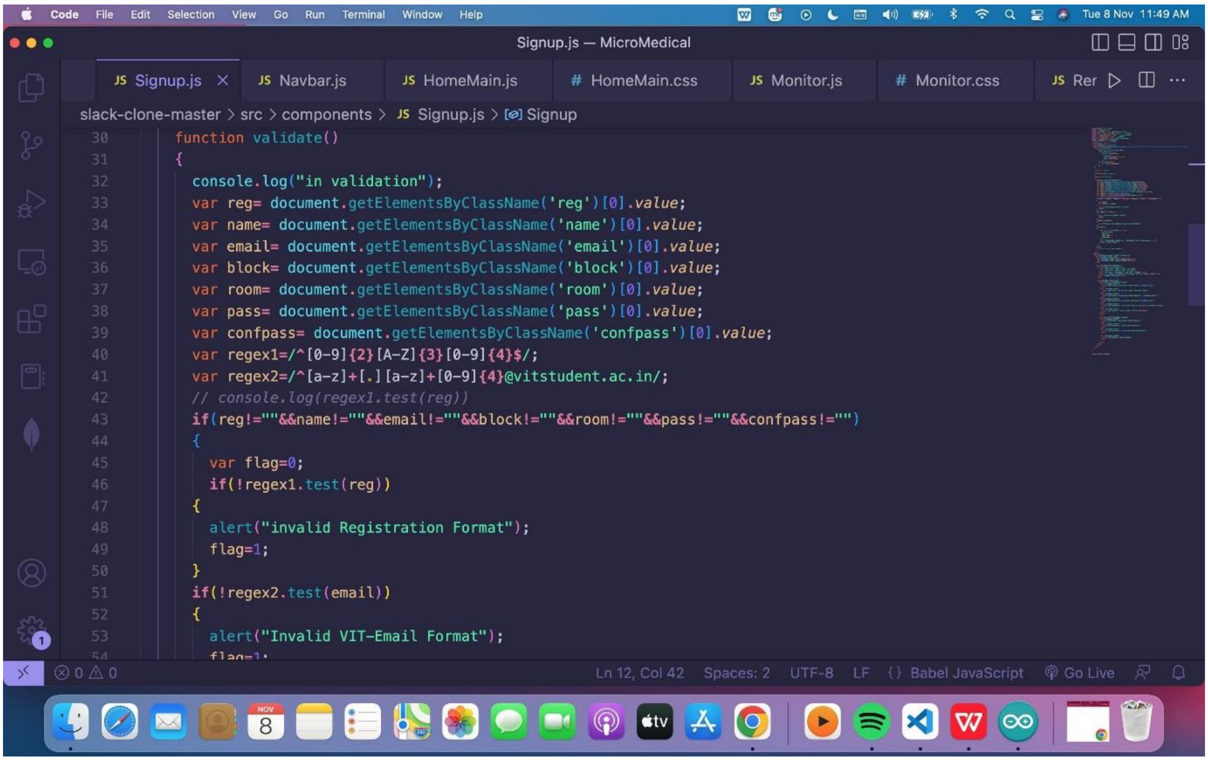


```
 // Clears the trigPin condition
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin HIGH (ACTIVE) for 10 microseconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echoPin, HIGH);
  // Calculating the distance
  distance = duration * 0.034 / 2; // Speed of sound wave divided by 2 (go and back)
  // Displays the distance on the Serial Monitor
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

// Firebase Error Handling And Writing Data At Specifed Path******************


if (Firebase.setInt(firebaseData, "/data", distance)) {    // On successful Write operation, function returns 1
             Serial.println("Value Uploaded Successfully");
             Serial.print("Distance = ");
             Serial.println(distance);
             Serial.println("\n");


             delay(1000);

     }

else {
    Serial.println(firebaseData.errorReason());
  }

}
```

➢ Sign Up and Login Code:

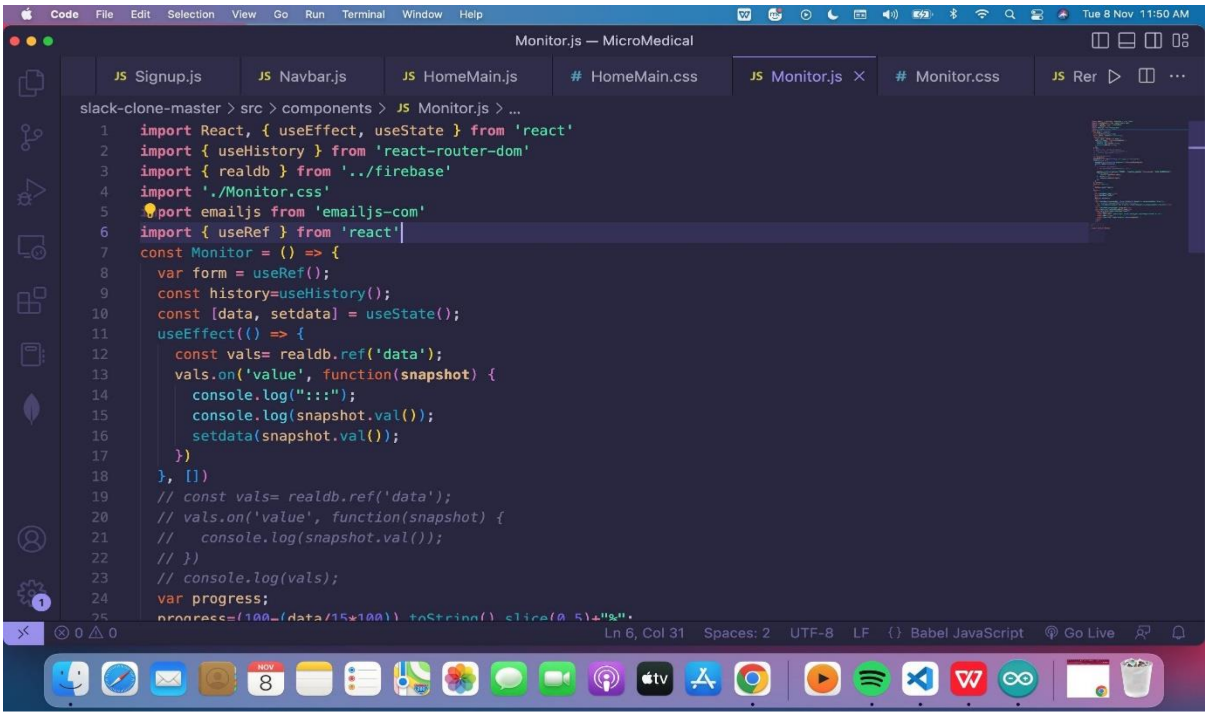


```
import React from 'react'
import './Signup.css'
import { Button } from '@material-ui/core'
import db, { auth, provider } from '../firebase'
import { useStateValue } from '../StateProvider'
import { actionTypes } from '../reducer'
import { useHistory } from 'react-router-dom'
import firebase from 'firebase'
// import db from '../firebase'
const Signup = () => {
 const history= useHistory();
  const [state,dispatch]=useStateValue();
    const signIn=()=>{
        auth.signInWithPopup(provider)
        .then(result=>{
            console.log(result);
            dispatch({
              type:actionTypes.SET_USER,
              user: result.user,
            })
        }).catch((error)=>{
            alert(error.message);
        })
    }
```

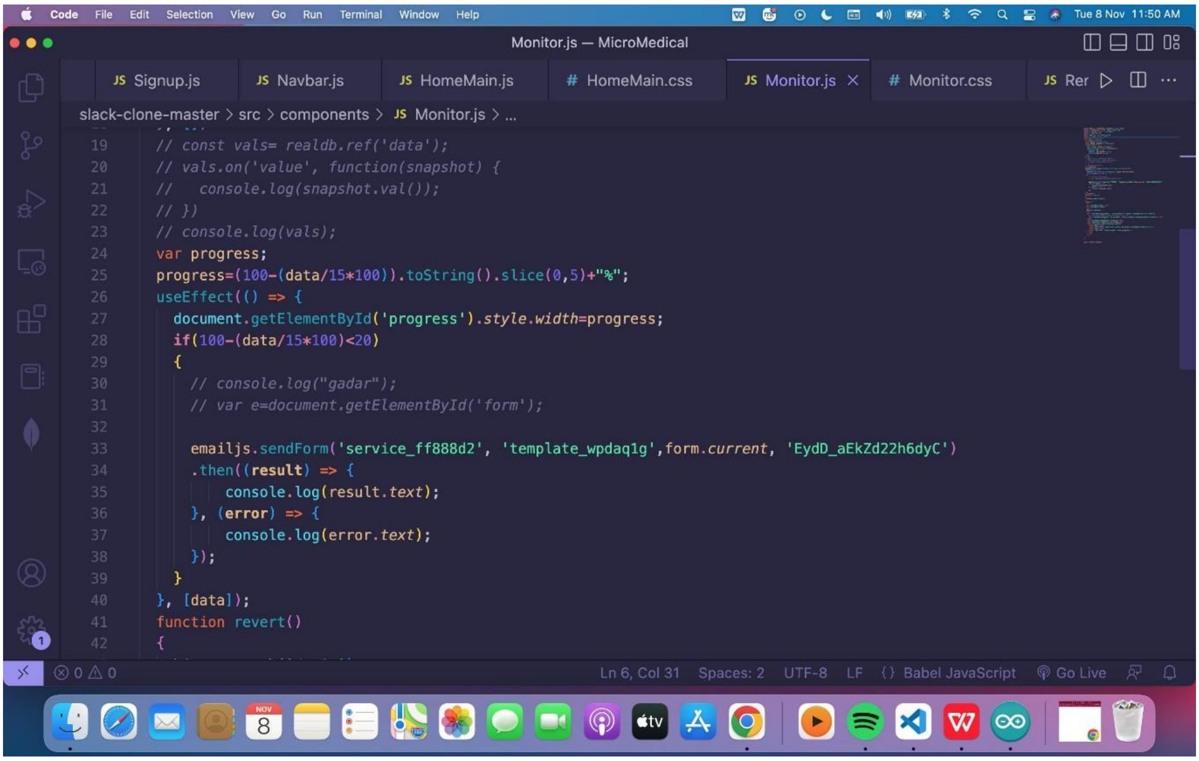


```
function validate()
{
  console.log("in validation");
  var reg= document.getElementsByClassName('reg')[0].value;
  var name= document.getElementsByClassName('name')[0].value;
  var email= document.getElementsByClassName('email')[0].value;
  var block= document.getElementsByClassName('block')[0].value;
  var room= document.getElementsByClassName('room')[0].value;
  var pass= document.getElementsByClassName('pass')[0].value;
  var confpass= document.getElementsByClassName('confpass')[0].value;
  var regex1=/^[0-9]{2}[A-Z]{3}[0-9]{4}$/;
  var regex2=/^[a-z]+[.][a-z]+[0-9]{4}@vitstudent.ac.in/;
  // console.log(regex1.test(reg))
  if(reg!=""&&name!=""&&email!=""&&block!=""&&room!=""&&pass!=""&&confpass!="")
  {
    var flag=0;
    if(!regex1.test(reg))
  {
    alert("invalid Registration Format");
    flag=1;
  }
  if(!regex2.test(email))
  {
    alert("Invalid VIT-Email Format");
    flag=1;
```

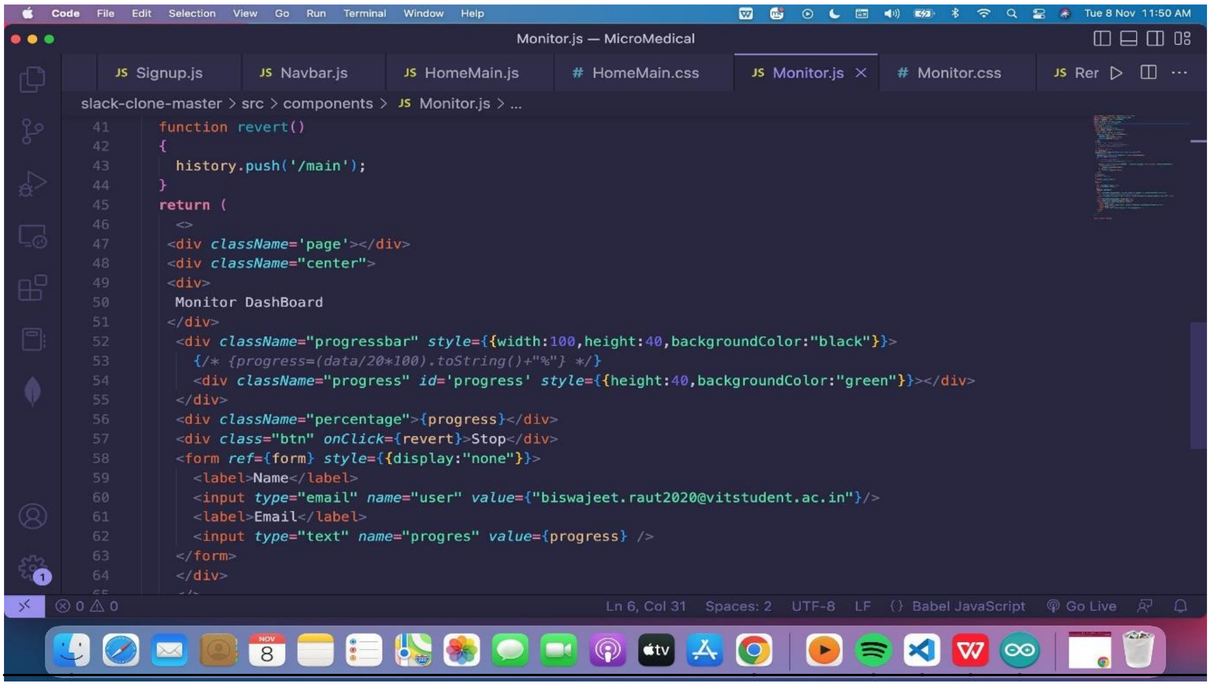
➤ Monitor.js Code:



```
import React, { useEffect, useState } from 'react'
import { useHistory } from 'react-router-dom'
import { realdb } from '../firebase'
import './Monitor.css'
import emailjs from 'emailjs-com'
import { useRef } from 'react'
const Monitor = () => {
  var form = useRef();
  const history=useHistory();
  const [data, setdata] = useState();
  useEffect(() => {
    const vals= realdb.ref('data');
    vals.on('value', function(snapshot) {
      console.log(":::");
      console.log(snapshot.val());
      setdata(snapshot.val());
    })
  }, [])
  // const vals= realdb.ref('data');
  // vals.on('value', function(snapshot) {
  //   console.log(snapshot.val());
  // })
  // console.log(vals);
  var progress;
  progress=(100-(data/15*100)).toString().slice(0,5)+"%";
```

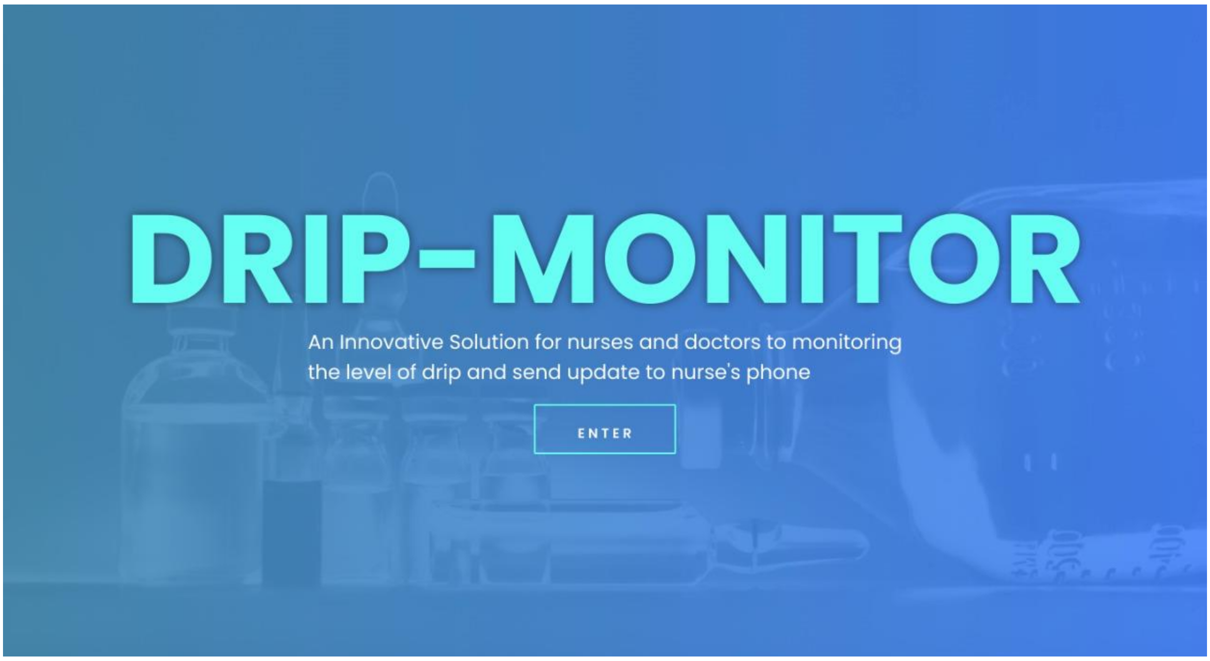


```javascript
// const vals= realdb.ref('data');
// vals.on('value', function(snapshot) {
//   console.log(snapshot.val());
// })
// console.log(vals);
var progress;
progress=(100-(data/15*100)).toString().slice(0,5)+"%";
useEffect(() => {
  document.getElementById('progress').style.width=progress;
  if(100-(data/15*100)<20)
  {
    // console.log("gadar");
    // var e=document.getElementById('form');

    emailjs.sendForm('service_ff888d2', 'template_wpdaq1g',form.current, 'EydD_aEkZd22h6dyC')
    .then((result) => {
        console.log(result.text);
    }, (error) => {
        console.log(error.text);
    });
  }
}, [data]);
function revert()
{
```

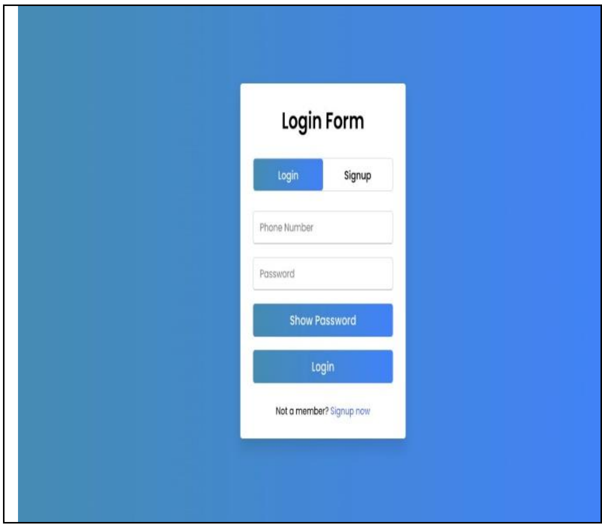


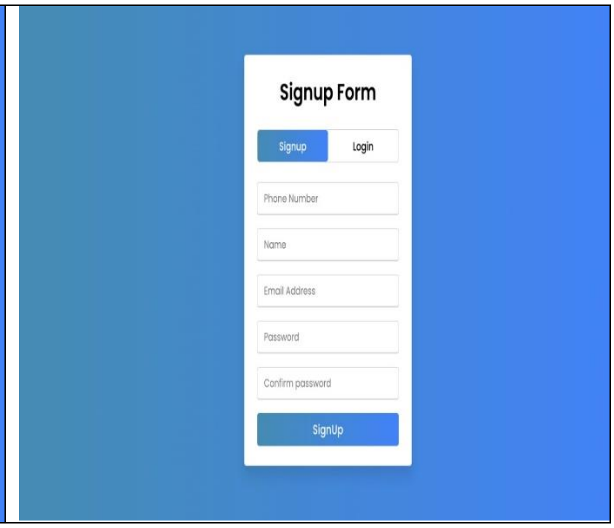
```javascript
function revert()
{
  history.push('/main');
}
return (
  <>
 <div className='page'></div>
 <div className="center">
 <div>
  Monitor DashBoard
 </div>
  <div className="progressbar" style={{width:100,height:40,backgroundColor:"black"}}>
    {/* {progress=(data/20*100).toString()+"%"} */}
    <div className="progress" id='progress' style={{height:40,backgroundColor:"green"}}></div>
  </div>
  <div className="percentage">{progress}</div>
  <div class="btn" onClick={revert}>Stop</div>
  <form ref={form} style={{display:"none"}}>
    <label>Name</label>
    <input type="email" name="user" value={"biswajeet.raut2020@vitstudent.ac.in"}/>
    <label>Email</label>
    <input type="text" name="progres" value={progress} />
  </form>
  </div>
```

➢ **Front Page of the Website:**

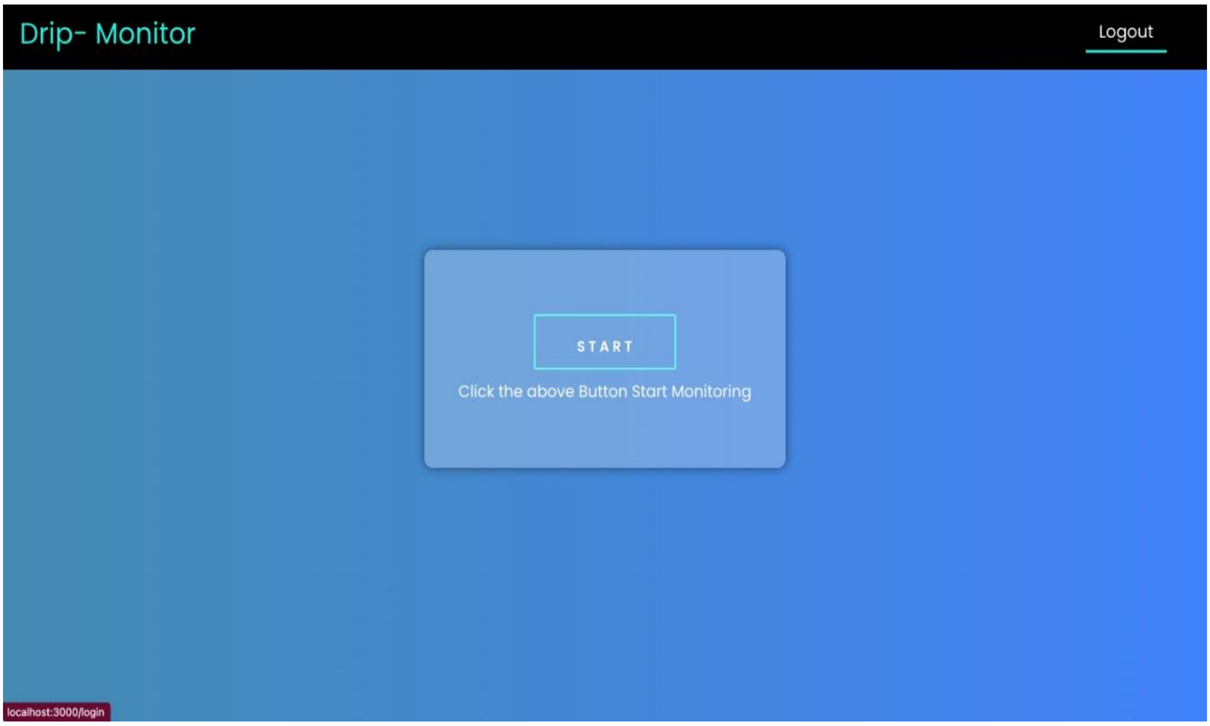


➢ **Sign Up and Login Page**

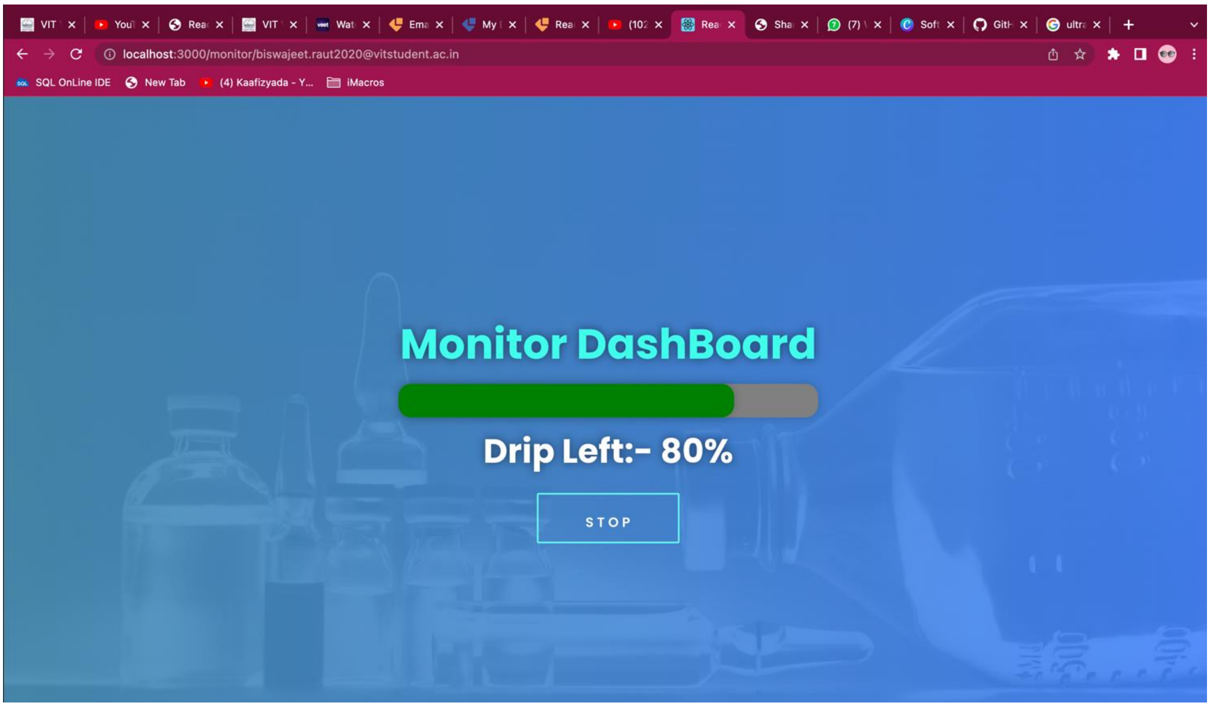




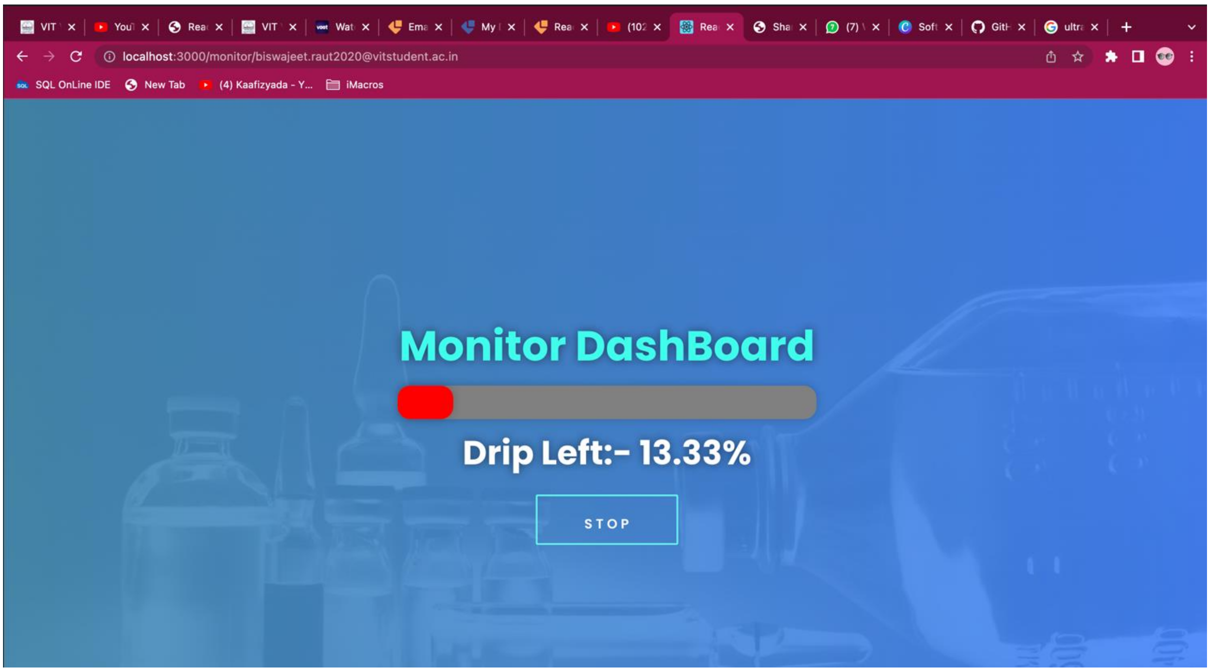
➢ After Successful login:



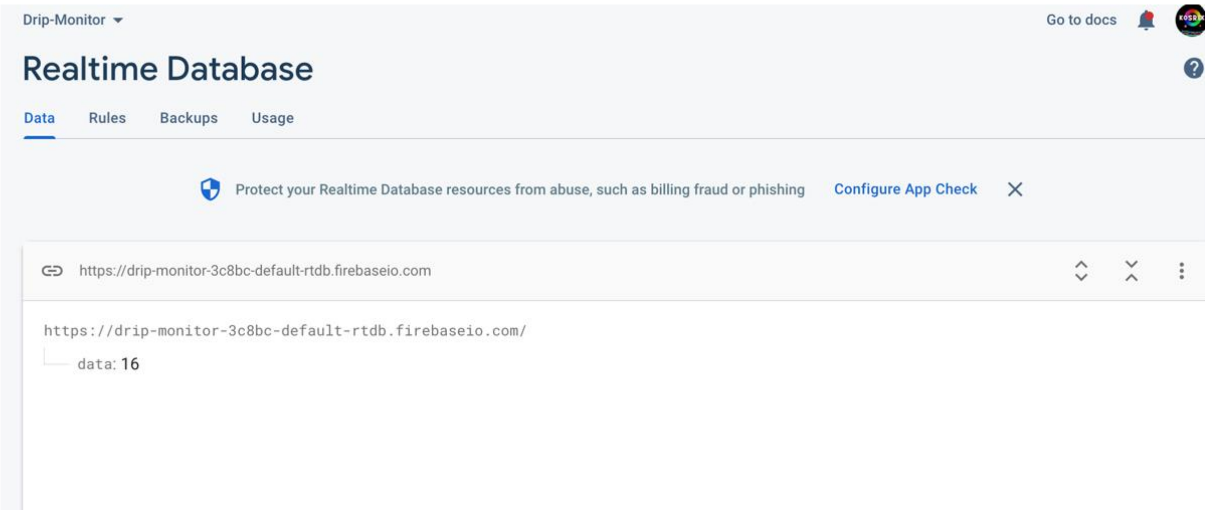
➢ Initialization:



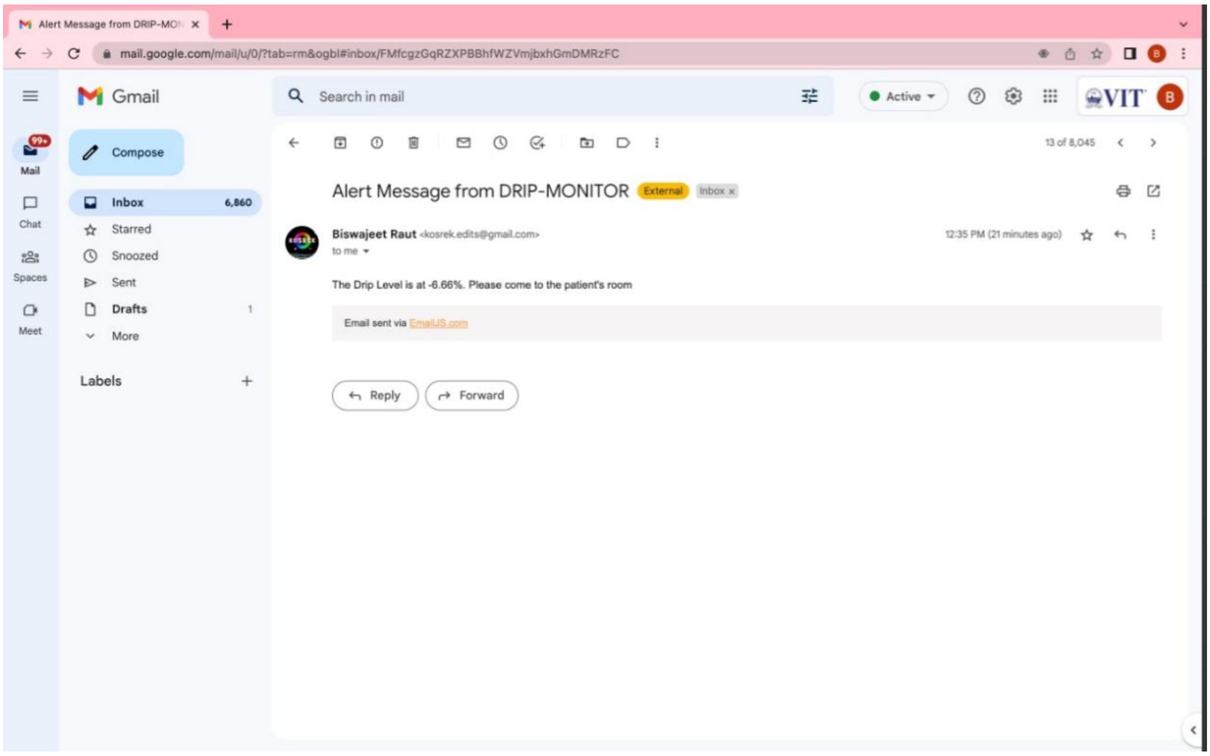
➢ When the level drops below the specified mark:



➢ Database Updation:



➢ Email Received:



➢ Hardware Implementation:

CONCLUSION

The Intravenous Drip Monitoring system using HCSR-04 is effective in monitoring the level of the IV solution given to the patients in the hospital, and this data can be reviewed by the hospital staff station which will enhance the retried facility for later reference. Using this system, the safety of patients is ensured as this system provides a high degree of reliability as the hospital staff gets a quick mail as soon as the level of the solution drops below a specific level. It helps in ensuring there are zero margins of error as improper administration of drip can lead to many problems. It also improves clinical efficiency, safety, and patient experience in hospitals and makes home care possible for many patients. The use of an ultrasonic sensor simplifies and expedites system implementation because it eliminates the need to calibrate the system for different fluids.

This system is highly cost effective as the HCSR-04 ultrasonic sensor and NodeMCU Development Kit combinedly are available for a meagre price of Rs. 500 against the expensive “Solenoid valves” used in the preexisting systems whose cost is in the range of Rs. 3,500-7,000. Thus, this is a cost effective and highly efficient system which would provide a great aid to the hospital workers in doing their jobs.

FUTURE WORKS

This system can be implemented in a hospital with many numbers of patients in a wing and this data can be represented more effectively with detailed information using graphical representation to the hospital staff using Machine Learning algorithms and the same idea can be changed to invoke automatic control, and be implemented using Internet of Things. To make this project a more visual one, a series of Light Dependent Resistors (LDR) can be integrated with the hardware whose color shall change according to the level of solution in the saline bottle.

REFERENCES

https://www.arduino.cc/en/pmwiki.php?n=Main/arduino BoardUno (Internet source)

https://dir.indiamart.com/impcat/solenoid-valves.html (Internet Source)

https://en.wikipedia.org/wiki (Internet Source)

S. N. Ibrahim, M. S. L. Hakim, A. L. Asnawi and N. A. Malik, "Automated Water Tank Filtration System Using LDR Sensor," 2016 International Conference on Computer and Communication Engineering (ICCCE).

Ashika A. Dharmale, Revati. Mehare, Ankita R. Bharti , Shweta R. Meshram, Prof. Swapnil V. Deshmukh, Computer Science and Engineering, PRMIT&R Badnera, Amravati, India ,"IOT Based Saline Level Monitoring & Automatic Alert System" on International Journal of Advanced Research in Computer and Communication Engineering Vol. 8, Issue 4, April 2019

B.Q Li, Y.F Ling, H.Y Zhang, S.Y Zheng: The Design And Realization Of Cherry Tomato Harvesting Robot Based On IOT. International Journal of Online Engineering, 12(12), 23, (2016).

S.Y Zheng, Et Al, and Digital Display Design of Ethnic Clothing of Nanling, In: Proceedings of the 4th International Conference on Mechatronics, Materials. Chemistry and Computer Engineering 2015 (ICMMCCE 2015), Xian, China, 2015, pp.2805-2808.

M.Arfan, M. Srinivasan, A. G. Baragur and V. Naveen, "Design and Development of IOT enabled IV infusion rate monitoring and control device for precision care and portability," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)

A. W. Setiawan, N. Yenas and D. Welsan, "Design and Realization of Low-Cost Wireless Remote Infusion Monitoring System," 2018 International Seminar on Application for Technology of Information and Communication

N. Giaquinto, M. Scarpetta, M. A. Ragolia and P. Pappalardi, "Real-time drip infusion monitoring through a computer vision system," 2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA).

F. Noor, M. Swaied, M. AlMesned and N. AlMuzini, "A Method to Detect Object's Width with Ultrasonic Sensor," 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE).

INTRAVENOUS DRIP MONITORING SYSTEM USING HCSR-04

SANDEEP RAJ
School of Computer Science and Engineering
Vellore Institute of Technology
Vellore,India
sandeep.raj2020@vitstudent.ac.in

BISWAJEET RAUT
School of Computer Science and Engineering
Vellore Institute of Technology
Vellore,India
biswajeet.raut2020@vitstudent.ac.in

BIDWAN RATH
School of Computer Science and Engineering
Vellore Institute of Technology
Vellore,India
bidwan.rath2020@vitstudent.ac.in

SHAUNAK DESHMUKH
School of Computer Science and Engineering
Vellore Institute of Technology
Vellore,India
shaunakravi.deshmukh2020@vitstudent.ac.in

DR. AJITHA D
School of Computer Science and Engineering
Vellore Institute of Technology
Vellore,India
ajitha.d@vit.ac.in

Abstract*— During recent years, we have seen much technological advancement which help to take better care of patient's health and assure them fast and safe recovery. The most basic item necessary is competent patient care in hospitals, as well as proper management of fluid and electrolytes. In hospital almost all patients, mainly ICU patients, have to regulate the volume of fluids and electrolyte into the bloodstream 24X7 which is done using drip. These drips need regular monitoring or changing to maintain constant flow of fluids or to prevent any infection to patients. Though this system is simple, overcrowded hospitals and the risk of deficiency of nurses can put patients at health risk which can deteriorate patients' health more aggressively or in worse case it can lead to death. Almost in all hospitals, nurses or the hospital staff are responsible for monitoring the drip level. But unfortunately, because of their hectic schedule, the observer may forget to change the bottle at the appropriate time. During the pandemic, hospitals were overrun with patients, and nurses were unable to do manual regular checks on the drip conditions and drip level of every patient, even after working extra shifts. Many patients even died due to not being able to get proper care from nurses. The next step in providing more effective and easy healthcare is to automate such vital procedures. To overcome this critical situation, we are proposing a NodeMCU base Intravenous Drip Monitoring System using HCSR-04 which eases the process of measuring and solves the issues of bubble formation in drips.

Keywords*— Firebase, ultrasonic sensor , NodeMcu devkit, , Wi-fi Module , Arduino IDE

I. INTRODUCTION

Intravenous therapy is a process of administering medicine into the body directly through the veins. It is the fastest way of delivering fluids and medications throughout the body as it utilizes the effective cardiovascular channel and its natural pumping forces. The working principle of IV is that a bottle filled with the desired fluid medication is hung at a level higher than the patient's body to provide the fluid with a pressure, generated by gravitational potential energy, to overcome the cardiovascular pressure. IV therapy is an easy and effective procedure which allows a drop-by-drop administration of medication. The process also has some innate limitations, like formation of tissue in the needle, rolling of patient on the tube or over the hand, can block the flow of the fluid and compromises with the patient's life in severe scenarios. The major bottleneck of the process is monitoring the medicine bottle. This is necessary so that the doctor or the nurse can know when to change the bottle. In general, the bottle is changed

when it is completely empty. The time it takes to empty is variable and depends on parameters such as medicine quantity used, changes in back-pressure due to systolic-diastolic and non-quantifiable dilation or contractions of veins. Because of the dynamic nature of this process, measurement based robust monitoring is necessary. Failure of this can lead to various medical complications like backflow of blood into the IV setup. In severe cases, if the bottle gets empty and it is not monitored for some time, it can also cause insertion of air embolism in the IV tube, which can be deadly and hence critical monitoring is required. Currently, this monitoring is done by the nurses and/or by the patient's attendant and in countries like India, they have a poor nurse to patient ratio. This project deals with design, fabrication and testing of an Intravenous Drip Monitoring System using HCSR-04, which is capable of precisely monitoring drip level in the saline bottles for the patients, and giving an alert message to the concerned doctor or nurse when the level falls below a specific reading as decided upon by the concerned doctor or nurse.

II. LITERATURE REVIEW

Nicola Giaquinto, Marco S carpetta and Mattia Alessandro Ragolia proposed "Real-time drip infusion monitoring through a computer vision system" in 2020 IEEE Explore. The method proposed in this system monitors intravenous infusion using Microwave Time Domain Reflectometry. A method based on deep learning computer vision for IV drip monitoring was proposed. This technique was found to be less invasive than other available solutions, which require a direct contact with the infusion kit, offering good accuracy performance anyway. The advantage of this system is that the the control system can be better in time consumption and the system can easily control the hardware.

Shiyong Zheng, Zhao Li and Biqing Li proposed "The design of liquid drip speed monitoring device system" in AIP Conference proceedings – 1864, 020123 in 2019. The main function of this design is to address the unattended patients in the process of having an intravenous drip, nonadjustable solution velocity and the insufficient treatment after having an intravenous drip. This design mainly adopts infrared photoelectric sensor. The SCM AT89C51 chip is a small and complex system, which includes hardware design and software design in this whole system. The advantage of this system is that the smallest system of SCM is small in size, simple in structure and is good in repeatability and high in security based on MCU

III. TECHNOLOGY USED

A. HARDWARE REQUIREMENTS

1.NodeMCU DEVKIT 1.0:

The NodeMCU Development Kit is based on ESP8266 and it integrates GPIO, PWM, IIC, 1- Wire and ADC all in one board. It has Arduino like hardware IO, Advanced API for hardware IO, which can dramatically reduce the redundant work for configuring and manipulating hardware. It has Nodejs style network API and Event-driven API for network applications, which facilitates developers writing code running on a 5mm*5mm sized M

2. UltraSonic Sensor (HC-SR04):

It is an ultrasonic sensor, also known as an ultrasonic transducer that is based on a transmitter and receiver and mainly used to determine the distance from the target object. The amount of time it takes to send and receive waves will determine how far the object is placed from the sensor. It mainly depends on the sound waves working on "non-contact" technology. The required distance of the target object is measured without any damage, giving you accurate and precise details. This sensor comes with a range between 2cm to 400cm and is used in a wide range of applications including speed and direction measurement, wireless charging, humidifiers, medical ultrasonography, sonar, burglar alarms, and non-destructive testing.

Working Voltage: DC 5V

3.Esp8266-01 Wi-Fi Module:

The ESP8266 ESP-01 is a serial to Wi-Fi breakout module with a built in ARM microprocessor that has 1MB of memory and 2 GPIOs brought out to the header for connecting to peripherals. It can be used as a serial to Wi-Fi bridge to add Wi-Fi capability to a project or it can even be programmed directly and used as a little stand-alone processor. It has full TCP/IP capability built-in. It uses a 32-bit RISC Tensilica Xtensa LX Processor running at 80MHz and a Flask.

IV. SOFTWARE REQUIREMENTS

1.FIREBASE: The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in Realtime to every connected client. When we build cross-platform apps with our Apple platforms, Android, and JavaScript SDKs, all of our clients share one Realtime Database instance and automatically receive updates with the newest data. The Firebase Realtime Database can be accessed directly from a mobile device or web browser; there's no need for an application server. Security and data validation are available through the Firebase Realtime Database Security Rules, expression-based rules that are executed when data is read or written. Firebase apps remain responsive even when offline because the Firebase Realtime Database SDK persists your data to disk. Once connectivity is re-established, the client device receives any changes it missed, synchronizing it with the current server state.

2.VSCODE: Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages and runtimes (such as C++, C#, Java, Python, PHP, Go, .NET). It is made by Microsoft with the Electron Framework, for Windows, Linux and macOS. In the Stack Overflow 2021 Developer Survey.

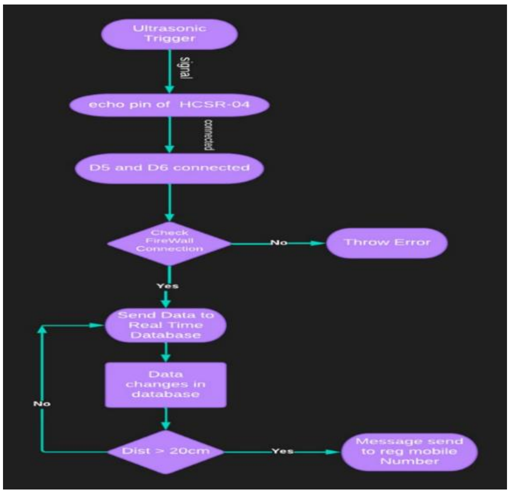
3.Arduino IDE: The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them. Programs written are called sketches. These sketches are written in the text editor and are saved with the file extension ".ino". The Arduino Software (IDE) uses the concept of a sketchbook: a standard place to store your programs (or sketches). Beginning with version 1.0, files are saved with a ".ino" file extension. Previous versions use the ".pde" extension

V. METHODLOGY

In brief explanation, we have first used a NodeMCU and connected it to ultrasonic sensor. The ultrasonic sensor have 4 pins. These pins are Echo, Trigger, Gnd, Voltage. The GND and Vol pins are connected directly with the GND and Vol pins of the NodeMCU. and the Echo and Trig pins are connected to D5 and D6 of teh NodeMCU respectively. An HC-SR04 ultrasonic distance sensor actually consists of two ultrasonic transducers.One acts as a transmitter that converts the electrical signal into 40 KHz ultrasonic sound pulses. The other acts as a receiver and listens for the transmitted pulses.When the receiver receives these pulses, it produces an output pulse whose width is proportional to the distance of the object in front.This sensor provides excellent non-contact range detection between 2 cm to 400 cm (~13 feet) with an accuracy of 3 mm.Since it operates on 5 volts, it can be connected directly to an Arduino or any other 5V logic microcontroller. So, after transimisson and recieving of the signals the data is passed to the D5 and D6 pins and then this data is being read by the Arduino code we have writetn in ARduino IDE. This code is connected to the hardware by the means of Wifi. We need to specify the wifi name and password in teh code by which is the data is being transfered. At the same time the data is being transfered to teh Firebase realtime database by means of database connection with auth specifications. This database is connected by the commands:
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.reconnectWiFi(true); So, after the connection is established, the data is being updated in the realtime database with a delay of 600ms. Then this data is being collected by the website code. And is showed in the database. The drip level is being calculated in the code and its less than 20 percent then it is being notified to the registered nurse's email id by using EMAILJS.

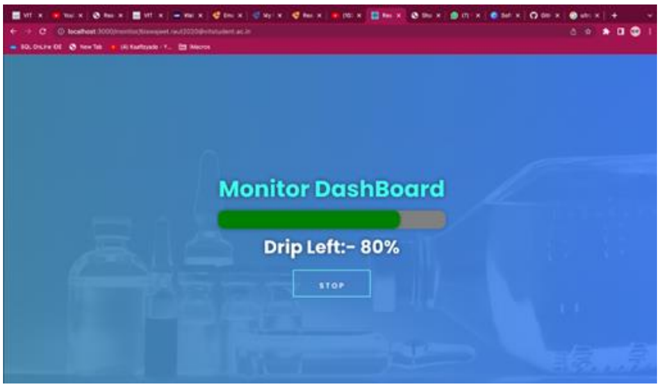
VI. IMPLEMENTATION

The proposed methodology is to provide help to hospital staff to keep a check on the drip level of the patient. So, we use NodeMCU and Ultrasonic sensor to detect and send the distance of the drip level from the sensor level. After the detection we connect the data with the Realtime database service provided by firebase (Google Cloud Services). This is data is constantly updated as the drip level changes. Now this data is being collected or fetched by the website and it shows the percentage of saline remaining and it sends a message to the hospital staff's email id.

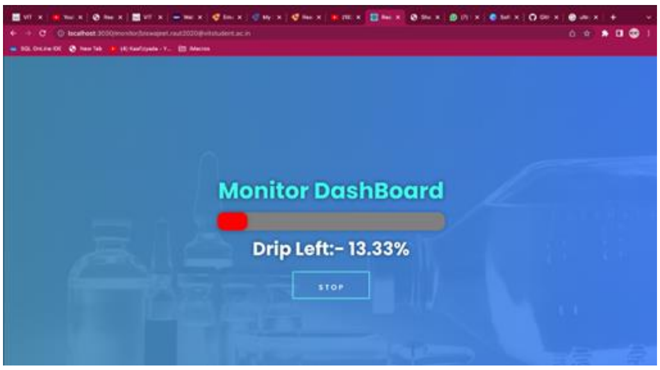


VII. RESULTS

After the proper configuration of the hardware system with the saline bottles at the hospital, the hospital staffs logs in in the Drip Monitoring Portal.



When the level of the saline water falls below the 20% mark, an email is sent to the staff on the email id through which the staff member has logged in the system. The emails are received continuously till the level of water in the saline water is increased above the 20% mark.



VIII. CONCLUSION

The Intravenous Drip Monitoring system using HCSR-04 is effective in monitoring the level of the IV solution given to the patients in the hospital, and this data can be reviewed by the hospital staff station which will enhance the retried facility for later reference. Using this system, the safety of patients is ensured as this system provides a high degree of reliability as the hospital staff gets a quick mail as soon as the level of the solution drops below a specific level. It helps in ensuring there are zero margins of error as improper administration of drip can lead to many problems. It also improves clinical efficiency, safety, and patient experience in hospitals and makes home care possible for many patients. The use of an ultrasonic sensor simplifies and expedites system implementation because it eliminates the need to calibrate the system for different fluids.This system is highly cost effective as the HCSR-04 ultrasonic sensor and NodeMCU Development Kit combinedly are available for a meagre price of Rs. 500 against the expensive “Solenoid valves” used in the preexisting systems whose cost is in the range of Rs. 3,500-7,000. Thus, this is a cost effective and highly efficient system which would provide a great aid to the hospital workers in doing their jobs

ACKNOWLEDGMENT

The team members would like to acknowledge the efforts of our project guide, Dr. Ajitha D , her continuous efforts have propelled us to achieve our ideas. We would also like to thank all of our friends who have given valuable inputs in the pursuit of this project. We would also like to thank our college: Vellore Institute of Technology, Vellore for providing us with the infrastructure to perform the project.

REFERENCES

[1] S. N. Ibrahim, M. S. L. Hakim, A. L. Asnawi and N. A. Malik, "Automated Water Tank Filtration System Using LDR Sensor," 2016 International Conference on Computer and Communication Engineering (ICCCE).

[2] Ashika A. Dharmale, Revati. Mehare, Ankita R. Bharti , Shweta R. Meshram, Prof. Swapnil V. Deshmukh, Computer Science and Engineering, PRMIT&R Badnera, Amravati, India ,“IOT Based Saline Level Monitoring & Automatic Alert System” on International Journal of Advanced Research in Computer and Communication Engineering Vol. 8, Issue 4, April 2019

[3] B.Q Li, Y.F Ling, H.Y Zhang, S.Y Zheng: The Design And Realization Of Cherry Tomato Harvesting Robot Based On IOT. International Journal of Online Engineering, 12(12), 23, (2016).

[4]] S.Y Zheng, Et Al, and Digital Display Design of Ethnic Clothing of Nanling, In: Proceedings othInternational Conference on Mechatronics, Materials. Chemistry and Computer Engineering 2015 (ICMMCCE 2015), Xian, China, 2015, pp.2805-2808.

[5] M.Arfan, M. Srinivasan, A. G. Baragur and V. Naveen, "Design and Development of IOT enabled IV infusion rate monitoring and control device for precision care and portability," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)

[6] A. W. Setiawan, N. Yenas and D. Welsan, "Design and Realization of Low-Cost Wireless Remote Infusion Monitoring System," 2018 International Seminar on Application for Technology of Information.

[7] N. Giaquinto, M. Scarpetta, M. A. Ragolia and P. Pappalardi, "Real-time drip infusion monitoring through a computer vision system," 2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA).

[8] F. Noor, M. Swaied, M. AlMesned and N. AlMuzini, "A Method to Detect Object's Width with Ultrasonic Sensor," 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE).