



# VOICE CONTROLLED VEHICLE USING MQTT SERVER

N Ayush Ubale  
Department of Computer Science  
Jain University  
Bangalore,India

Pranavya M U  
Department of Mangement  
Jain University  
Bangalore,India

Poli Guha Neogi  
Department of Computer Science  
Jain University  
Bangalore,India

Hardik Jethava  
Department of Computer Science  
Jain University  
Bangalore,India

**Abstract** - The voice-controlled vehicle was created to make human work easier, since we live in an artificial intelligence-driven world where robots perform many tasks. The human voice is used to drive the vehicle. A stable android mobile application built with android studio software transmits the speech. It's essentially a Wi-Fi link. Using the mobile application, we can operate the vehicle with our voice from anywhere. The NodeMCU IoT framework is free and open source. It includes firmware for the ESP8266 Wi-Fi module, a Espressif Systems SoC, and ESP-12 module hardware. With the application that has been created, human work may become simpler. Since the vehicle will be linked to Wi-Fi, we will be able to access it from any location in our project. He or she will use the Android application to send commands or voice commands such as forward, backward, left, right, left forward, left backward, right forward, right backward, and right forward, right backward. The pins have been connected to the NodeMCU esp8266, and the code to control the car has been written. When an object or vehicle inhibits the car's movement, an ultrasonic sensor is used to stop it. The voice is recognised and forwarded to the HiveMQ server, where it is processed. HiveMQ's MQTT broker is optimised for cloud native deployments to take advantage of cloud resources. The use of MQTT reduces the amount of bandwidth used to transport data across the network. Low overall operating costs are a result of efficient IoT solutions. The hivemq server receives the command from the Android application and sends it to the NodeMCU microcontroller, which programmes the code to drive the vehicle. The voice-controlled robot vehicle project has military, surveillance, and human applications in scope. It's a voice-activated wireless robot vehicle. The project's main goal is to guide the robotic vehicle to a specific location. In addition, the project's main goal is to use voice to control the robot. It is now possible to have human-robot interaction. The aim of a voice-controlled Robot is to listen to the user's commands and act on them.

**Keywords**— Hive MQTT Server, Voice Controlled Vehicle, Android Application, Node MCU, IOT.

## I. INTRODUCTION

The Internet of Things (IoT) is a concept that connects all devices to the internet and enables them to communicate with each other. Devices and artefacts with built-in sensors are connected to an Internet of Things network, which collects data from a variety of sources and applies algorithms to share the most valuable data with applications personalized to particular needs. Without the need for human intervention, machines will communicate among each other, resulting in rapid and more efficient development. The Internet of Things (IoT) was primarily most relevant to industry and infrastructure, where its deployment is often referred to as machine-to-machine (M2M), but the emphasis is now on bringing smart devices into our daily lives, making it relevant to almost all. The synchronization of an IoT network is its support network. To link and share data between devices, we use a variety of methods. Wi-Fi, Bluetooth, Ethernet, and Long-Term Evolution are all examples of these approaches (LTE). Edge Computing is the most recent development, in which more processing is performed on-device and less and lesser valuable data is sent to the cloud in order to save money.

**Human voice** is used to operate the voice-controlled vehicle. The **android smartphone application** transmits the speech. NodeMCU ESP8266 Microcontroller produced this voice-controlled vehicle. This microcontroller allows for excellent RF and power output. Using the mobile application, we can command the vehicle with our voice from anywhere. The **MQTT Protocol (Message Queue Telemetry Transport)** is used to interact. This is a lightweight protocol that enables many devices to connect to the MQTT Broker, a distributed server, and publish to or write to user-defined topics. If a computer subscribes to a subject, the broker can send it an alert if the topic changes. MQTT is ideally suitable for applications



requiring low latency, strong power constraints, or poor Broadband connections. The voice-controlled vehicle was created as an example of how voice-enabled technology can assist in improving overall individual interactions and simplifying daily human activities.

Since cell phones are a part of our daily lives, a lot has changed. For example, phones are no longer just devices for calling and receiving phone calls; you can use them to monitor your home or car, shop, do creative work, or run a company. These tasks become much simpler and quicker with voice control. It can be used when driving, vacuuming, cooking, or by people with reduced hand mobility. People who live quickly know that making a short call takes a lot less time than typing a long email and awaiting a response.

## II. OBJECTIVES

The primary goal is to allow a voice-controlled car to navigate to a specific location. Furthermore, ultimate aim is to monitor the robot via voice commands. It is now possible to have human-robot interaction. The aim of **voice-activated technology** is to allow people to control their environment through their voices the user's instructions must be listened to and followed by the robot. The proposed system is made up of two parts: a transmitter and a receiver, all of which are powered by a microcontroller and a battery. We can monitor the robotic vehicle with the help of a smartphone and a human voice. The project's aim is to use voice commands to operate a robotic vehicle.

## III. LITERATURE SURVEY

M Saravanan [1] developed "Arduino Based Voice Controlled Robot Vehicle" (October 2020) The main goal of this device is to create a robot vehicle that can be powered by a person's voice order. These systems are commonly referred to as Speech Controlled Automation Systems (SCAS). The above-mentioned device is a prototype of our design. The concept is to build a robot that will be controlled by voice commands. A cell phone is used to operate the robot; there are several articles that demonstrate the contact between a robot and a smart phone. For remotely automating the robot, a smart phone is an excellent interface. It has a lot of functionality that can be useful. The specified task is carried out in this design using an android application and a microcontroller. Bluetooth technology facilitates communication between the software and the robot. The module will receive the commands that are sent over the channel. The aim of a voice-controlled robotic vehicle (VCRV) is for it to listen to and respond to the user's commands.

H. Jagadish Kumar [2] wrote "Voice Controlled Car using Arduino and Bluetooth Module" (December 2019). The objective of this report is to build a voice-activated car that reacts speech commands. Enhancements in the areas of disturbance and range handling are, nevertheless, needed.

Simple voice commands such as left, right, forward, back, and stop are used to navigate the vehicle. These signals are sent to the Bluetooth module by an Android application. A Bluetooth module and a control device are used to record and analyse the voice input. In this suggested method, they conducted research on the different control type configurations for robots. It shows that real-world manuscripts can be effectively researched and replicated using only speech (human voice) as a means of control. The aim of this lookup is to develop a basic robotic hardware implementation so that this structure can concentrate on Bluetooth smart grids. When the app is running, a transmitter on the smartphone is used to identify user voice commands. The app characterizes commands and translates voice to text using Google's speech-recognition technology. The text would then be transmitted through Bluetooth to the receiver component.

Aditya Chaudhry [3] implemented "Arduino based voice-controlled Robot" (June 2019) This research paper proposes a method that focuses on the idea of controlling a robot with a voice signal. The voice control robot is merely an example of how to control the movements of a simple robot using common voice commands. In this system they have used ATmega3898P It's a microcontroller on a single chip. It has an 8-bit RISC processor core based on Harvard architecture. To monitor a robotic vehicle with our voice, we used a very reliable interface. First, all user signals are converted to script using Google's speech to text converter, that is built into the app. The command's text form will be sent to the robotic car's Bluetooth module. This Bluetooth module enables the car's microcontroller to the Android app for transmitting data. When the micro-controller receives the text signal, it directs the robotic car to drive in the appropriate direction.

Parichart Leechor [4] wrote "Operation of a Radio-Controlled Car by Voice Commands" (March 2010) The purpose of this research is to drive a radio-controlled car via voice commands (RC car). The experiment involved sending a user's voice commands to a device, which then converted them into digital data. Thereafter, the data output was converted into radio signal commands. Finally, an RC car was functioned using radio wave commands. The first feature is voice command recognition, for that they have adopted the Hidden Markov Model (HMM), which is one of multiple voice recognition technologies. The second part is the hardware, which involves a system dual port and RC car communication. To commence, we use signal processing theory to convert the human voice into a system digital signal. After that, the digital signal is converted to radio waves and used to power an RC vehicle.

Vipul Mehta [5] gave "Robot Controlled Car Using Wi-Fi Module" (May 2016) In this review, we'll glance at using a Wi-Fi module and an Android phone program to command a robot-controlled vehicle. We'll also demonstrate how to operate the appliances without an Android phone by sending a regular SMS. The aim of this paper is to show how to operate a robot-controlled vehicle using a Wi-Fi module and an Android Smart



Phone application. It also demonstrates that the devices can be operated by sending a regular SMS even if the user does not have an Android phone. This role can be easily modified to include a covert agent camera that streams the recordings to the client through Wi-Fi. Instead of the usual lithium-ion battery, the venture would use sunlight-based batteries.

Mrumal.K. Pathak [6] wrote “Robot Control Design Using Android Smartphone” (February 2015) The objective of this project is to build a robot which can be powered by an Android phone. An Android phone is used to command the robot using Bluetooth. In this design, the Android phone works as a remote control for the Robot. The overall system is connected to a microcontroller. Bluetooth module and DC motors are interfaced to the microcontroller. The Bluetooth module sends the information from the Android phone to the controller. The controller is in charge of the robot's DC motors. In order to execute the project, the controller is loaded with a programming language in the Embedded "C" language. The goal of this article is to develop exceptional functional android stages using less complex and confusing robot equipment. This paper describes how to monitor a robot with a handheld using Bluetooth communication, as well as some of the core concepts of Bluetooth technology and the compact and robot segments. It shows an analysis of robots enabled by a smartphone by moving the robot forward, backward, left, and right with an Android software like Micro controller or Bluetooth.

Shubh Srivastava [7] contributed “Voice controlled robot car using Arduino” (May 2020) The project's goal is to build a voice-controlled robotic car. Arduino microcontroller, motor drivers, and a Bluetooth module are used to power the device. Arduino is a piece of open-source hardware that can be used to create digital gadgets. A Bluetooth module is used to capture and read voice commands, allowing the control unit to communicate with the Bluetooth device. The controlling remote is a Bluetooth-enabled smart android smartphone. It is given a simple voice activated robotic vehicle. It constitutes of a mobile device that accepts voice commands and transmits them through Bluetooth to the Bluetooth module HC05. At a certain point, the module transfers the order to information, and the character sequence is sent to the Arduino for processing. As a method of decoding the string, the Arduino microcontroller implements additional tasks. The commands are sent to the generator, which powers and drives the attached motors.

#### IV. PROPOSED SYSTEM

Robotics are abundant in today's world; they minimise human effort and assist us in completing tasks more quickly and intelligently. The **NodeMCU ESP8266** is used in this project to build a voice-controlled robot vehicle. The NodeMCU ESP8266 is an open source IoT framework. The android application is used to steer the car. Recognizing a human voice emitted via a protected Android programme, the vehicle may

perform the functions. The car will drive forward, backward, left, and right using ultrasonic sensors, and further stop when an obstacle or some other vehicle gets in the way.

The main purpose of the research is to navigate a voice-controlled car to a particular route. Furthermore, the project's main aim is to monitor the robot using voice commands. It is now possible to have **human-robot interaction**. The aim of voice-activated technology is to allow people to control their environment through their voices the user's instructions must be listened to and followed by the robot. The proposed system is made up of two parts: a transmitter and a receiver, all of which are powered by a microcontroller and a battery. We can monitor the robotic vehicle with the help of a smartphone and a human voice. The project's aim is to use voice commands to operate a robotic vehicle.

The NODEMCU esp8266 Controller was used to develop a voice-activated robot vehicle. Forward, backward, left, and right commands are used to accelerate the vehicle. Indeed, using ultrasonic sensors, stop when any vehicles interfere. It is essentially linked to a Wi-Fi network. The android smartphone application gives the instructions. The wi-fi network in the android app is used to power the vehicle. Using the Android programme, the wi-fi robot car can be quickly transported from one location to another. Using wi-fi network technology, we can make the car perform a variety of tasks.

The project's main goal is to steer the **robotic vehicle** into the desired location. In addition, the project's main goal is to use voice control to control the robot. Human-Robot Interaction is now possible. The system is also cost-effective and time-saving, requiring less manual labour. It is also fast, simple, and convenient, assisting disabled and elderly people. The purpose of a voice-controlled Robot is to listen to and respond to the user's commands.

The design of the proposed system is depicted in **Figure 1** below, as well as the interaction between the user and the car through voice commands.

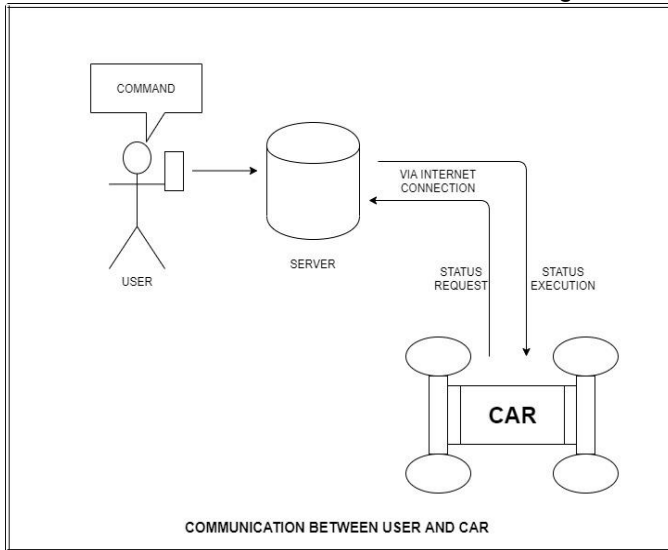


Figure.1 architecture diagram of the proposed system

### A. Modules Components

The aim of this proposed voice-controlled software is to A mobile robot that can be operated by voice commands is known as a robot vehicle. The **speech recognition programme** on an Android phone will recognise voice commands such as 'Forward,' 'Stop,' 'Left,' 'Right,' and 'Back,' among others. The robotic car's working mechanism is based on data sent from the phone to the robot.

**Microcontroller ESP8266:** Arduino model boards are widely used. The ESP8266, however, has a range of advantages over Arduino programming boards. The ESP8266 is a low-cost on-chip microcontroller with low energy consumption. It comes with dual-mode Bluetooth and Wi-Fi pre-installed. It is designed to provide flexibility, reliability, and robustness in a wide range of applications. MP3 decoding, voice encoding, and audio streaming are only a few of the applications for this microcontroller. This microcontroller can easily achieve the best RF and power output. Since the ESP8266 has a USB port, it can be considered a plug-and-play device.

The below **figure 2** illustrates the circuit diagram of our system. It shows how the pins and other devices connected to each other to control the voice control car in an efficient way.

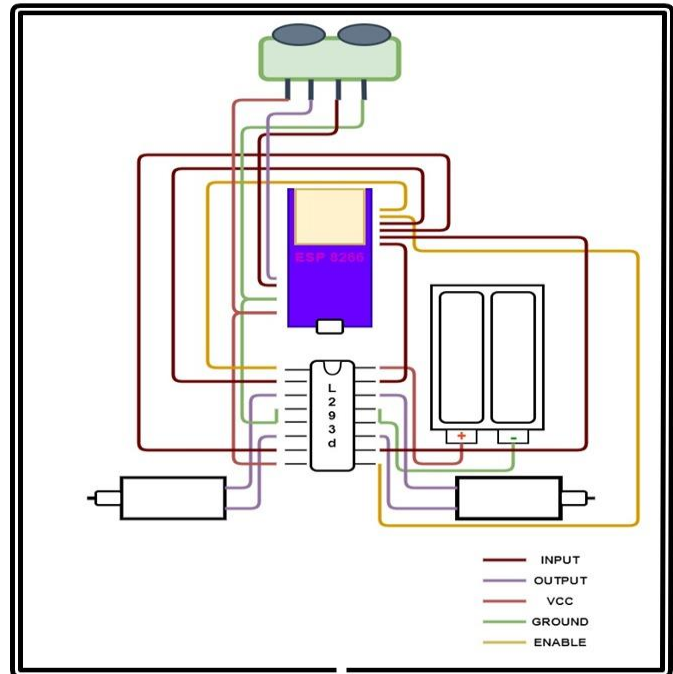


Figure.2 circuit diagram of our system.

**Motor Driver L293D:** High-current quadruple half-H generators, the L293 and L293D. The L293 can provide input and output drive currents of up to 1 A at voltages ranging from 4.5 to 36 V. The L293D is capable of asynchronous drive currents of up to 600 mA at voltages ranging from 4.5 to 36 V. Both systems are intended to drive voltage levels such as relay stations, solenoids, DC and bipolar stepping motors, and other high-current/high-voltage loads in parallel connection applications.

The below **figure 3** shows the motor driver L293D and the pins of it where we connect the other devices.

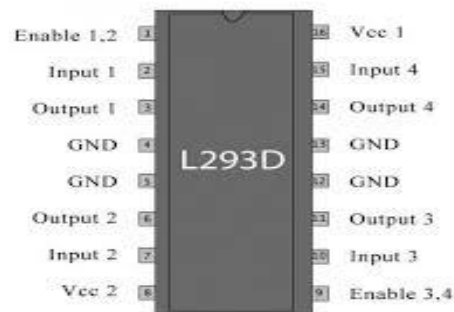


Figure.3 Motor Driver L293D

**NodeMCU (ESP8266):** NodeMCU is an open access LUA-based framework for the ESP8266 Wi-Fi chip. For exploring features with the ESP8266 chip, NodeMCU configuration comes with ESP8266 Development board/kit, i.e. NodeMCU



Development board. Anyone can modify, alter, or build their electronics because NodeMCU is an open source software. The NodeMCU Dev Kit/board includes the ESP8266 Wi-Fi enabled chip. The ESP8266 from Espressif Systems is a minimal Wi-Fi chip that relies on the TCP/IP protocol. Further data on the ESP8266 can be found at ESP8266 Wi-Fi Module.

The below **figure 4** illustrates the model of NodeMCU (ESP8266) where all the sensors and other IOT devices are connected using jumper cables.

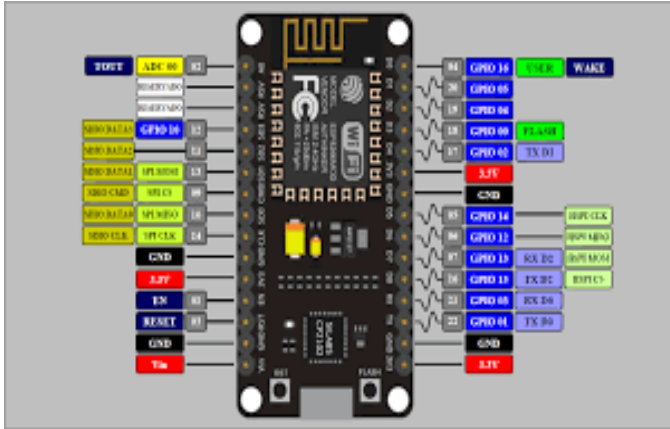


Figure.4 NodeMCU (ESP8266)

**Ultrasonic Sensors:** The ultrasonic ranging module HC - SR04 has a pseudo measurement range of 2cm to 400cm and a varying precision of 3mm. The modules include ultrasonic transmissions, receivers, and a communication module. The following are the work criteria in our system: The Module sends eight 40 kHz pulses and uses an IO trigger for at least a 10us high level signal to detect whether there is a pulse signal back. The longer it takes for ultrasonic to travel from sender to receiver is known as the high-performance IO range. (high-level time / sound velocity (340 M/S) / 2 = test distance.

**Figure 5** demonstrates ultrasonic sensors that assist the vehicle in avoiding collisions or injuries.



Figure.5 Ultrasonic Sensor

**Arduino Nano:** The Arduino Nano, based on the ATmega328 microcontroller, is a lightweight, feature-rich, and breadboard-

friendly board (Arduino Nano 3.x). It has similar characteristics to the Arduino Duemilanove, but it comes as a separate package. It only has a DC power socket and connects to the system through a Mini-B USB cable or perhaps a standard USB cable.

The below **figure 6** states Arduino Nano it is similar to NodeMCU but it has separate kit.

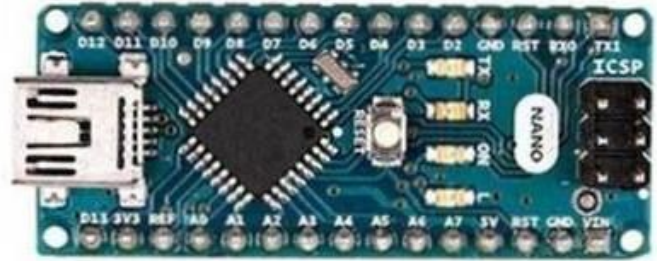


Figure.6 Arduino Nano

**Rechargeable 9 voltage Battery:** The nine-volt battery, also known as even the 9-volt battery, was first used in transistor radios in the early 1960s. It has a rectangular prism design with rounded edges and a mirrored snap connector on the end. Common nine-volt battery variants include main carbon-zinc and alkaline chemicals, primary lithium iron disulfide chemistry, and rechargeable nickel-cadmium, nickel-metal hydride, and lithium-ion batteries. A snap connector on one end of the battery links both interfaces. The smaller circular terminal serves as the positive contact, whereas the hexagonal or octagonal terminal serves as the destructive contact. But for the narrower plug, the battery has the same connectors as the load unit.

The below **figure 7** shows the 9-voltage battery it is used for power supply for the system.

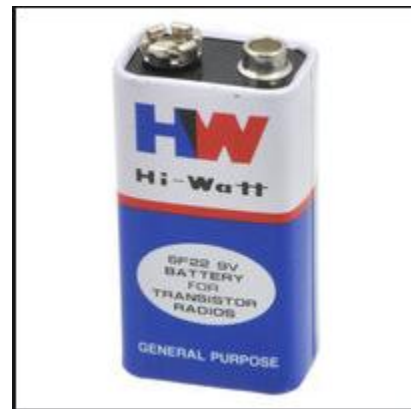


Figure.7 9-Voltage Rechargeable battery

*B. Software Components*



Any sensor, electronics, or software linked to the internet that can be used remotely and share data is considered Internet of Things technology. For improved functionality, technology often operates in parallel.

**Android Studio:** The official Integrated Development Environment (IDE) for creating Android apps is Android Studio, which is based on IntelliJ IDEA. On side of IntelliJ's versatile code editor and developer tools, Android Studio provides a unique Gradle-based build framework, a swift and feature-rich emulator, a centralized platform in which you can code for all Android devices, and Implement Improvements, which allows you to apply code and resource changes to your running app without reinstalling it.

The below **figure 8** illustrates the android studio application where the voice commands is converted to text and the commands are been transmitted to the vehicle module.



Figure.8 9 Android Studio Application

**Arduino IDE:** The Arduino Integrated Development Environment (IDE) is a bridge system for Windows, macOS, and Linux published in C and C++ functions. This is used to write and upload computer code to Embedded system boards and several other entrepreneurship development boards with 3rd-party core support. The source code of the IDE is protected by the GNU General Public License, edition 2. To endorse the languages C and C++, the Arduino IDE employs code generator structuring laws. The Connectivity project contains a library function in the Arduino IDE that includes various of standard input and output procedures. To launch the simulation and run the main programmed loop, app code only requires two basic calculations, which are assembled and linked with a configured stub main () into an executable file executive programmed using the GNU toolchain, which is updated to reflect IDE distribution. The avrdude software is used by the Arduino IDE to convert source codes to a text file in hexadecimal encoding, which is then loaded into the Arduino board's chipset by a loader. Avrdude is the default sharing tool for uploading code to approved Arduino boards.

The below **figure 9** shows the Arduino IDE where the code is been written in Embedded C Language to run to voice control vehicle.



Figure.9 Arduino IDE

**HiveMQTT Server:** Iot systems can generate a huge amount of data. Choosing a platform that can transport IoT data over the network and cloud platforms is crucial. The MQTT broker in HiveMQ is equipped for cloud native installations. Through use of MQTT decreases the amount of available bandwidth usage for data transfer. More efficient IoT technologies lead to lower net operating expenses. Inside any IoT approach, connecting and transmission of data to and from technology is a major task. HiveMQ follows the MQTT IoT standard procedure to securely connect any computer to a cloud service. Your clients demand IoT products to react immediately. HiveMQ uses a push technology configured for Iot devices to transmit and receive data from connected devices.  
PORT NO:1883

The below **figure 10** shows the HiveMQTT Server where it acts like a bridge between the user using the voice commands through the android application and the vehicle.



Figure.10 HiveMQTT Server

## V. WORKING SYSTEM

This system is designed to allow a person to control a car using only their voice. It is an internet of things-based voice control car that uses the HiveMQTT server as a connection between the

Android application and the Arduino IDE or the cod where the car is powered. We used a variety of IoT devices to create this model, including the NodeMCU ESP8266, Arduino Nano, a car model, motor driver L293D, 9-volt batteries, and ultrasonic sensors. Jumper cables were used to link all of these units.

The below **figure 11** shows the initial stage of deployment of the model where all the iot devices and sensors are been interconnected using jumper cables.

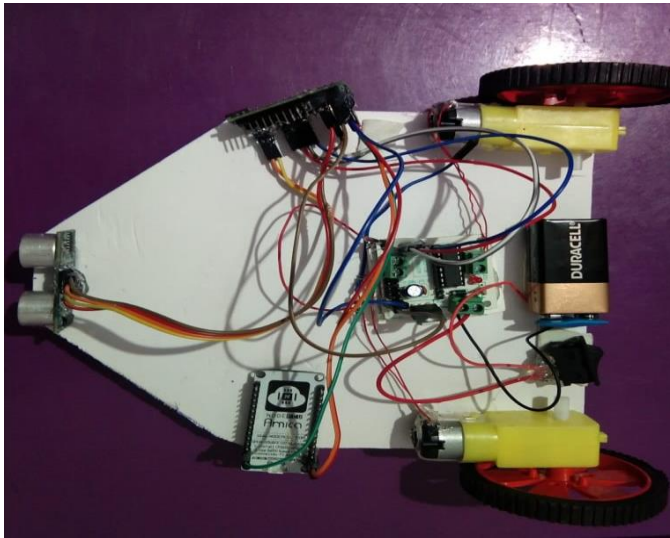


Figure.11 Initial stage of Implementation prototype

First, we used the Android Studio programme to build an Android application. The primary motivation for developing the application is that in previous frameworks, predefined applications such as Blynk dashboards and MQTT dashboards were used. The main drawbacks of these pre-existing applications are that if a consumer wishes to monitor a car, he or she can do so by understanding the model's topic name. As a result, we built an Android app. This application uses Google Speech Recognition to convert the voice into text and sends the order to the MQTT server.

The command is sent to the MQTT server once it has been processed. The MQTT server serves as a link between the Arduino IDE and the Android app. The MQTT client examines the command received by the android app and sends it to the MQTT server for processing. The command is tested by the MQTT server. Confirm that the order is right. The voice of someone speaking Forward, backward, left, right, and stop commands will be translated to F, B, L, R, and S. Since the MQTT server functions as a cloud database, it saves and processes commands before sending them to the Arduino IDE, where the code to control the car is published.

The below **figure 12** shows the working of HiveMQTT server as it acts like a bridge between the android application and the vehicle.

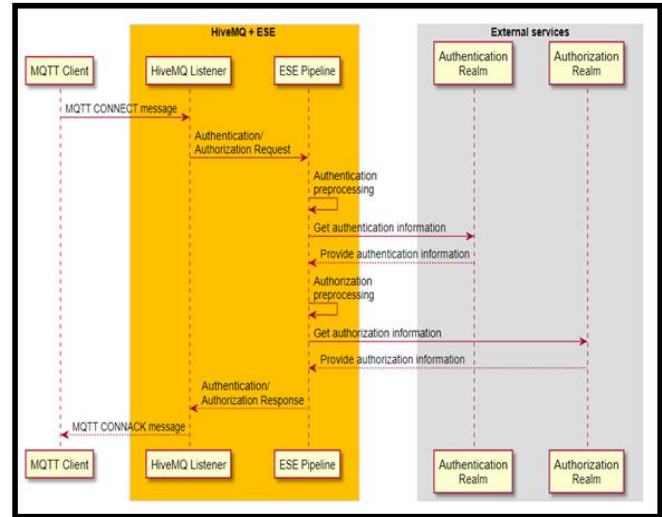


Figure.12 Working of HiveMQTT Server

The Arduino IDE controls the car whenever the request is sent to it, while the NodeMCU ESP8266 burns all of the code. The code has been deployed in the NodeMCU ESP8266 since it is the main component. The appropriate control is activated, and the vehicle moves. We can also use the MQTT dashboard to monitor the car by using buttons to send commands through the dashboard. We can provide power to this vehicle using our power banks or a 9-volt battery. The NodeMCU is used to attach the car's wheels and other devices such as ultrasonic sensors. The ultrasonic sensors prevent collisions by detecting objects or vehicles within a given range and stopping the vehicle to avoid a collision.

The below **figure 13** shows the iot devices and sensors connected to each other.

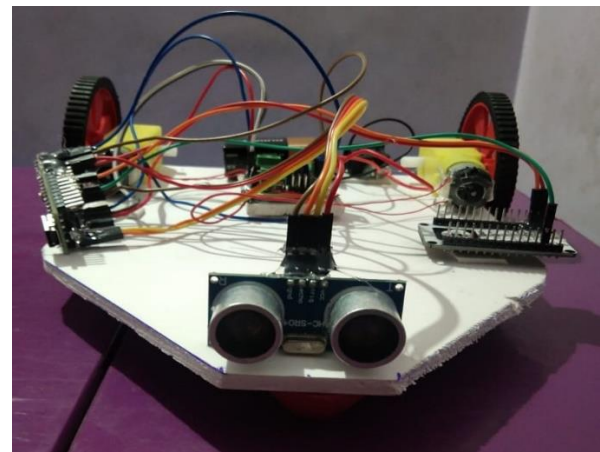


Figure.13 Initial stage of Implementation prototype



The primary goal of this project is to use human voice to drive the vehicle. The Android framework can be used to monitor it. Wi-Fi connects the android app to the car module. The voice control car can be used in a variety of settings, including military, personal use, and for the elderly and the primary goal of this project is to use human voice to drive the vehicle. The Android framework can be used to monitor it. Wi-Fi connects the android app to the car module. The voice control car can be used in a variety of settings, including military, personal use, and for the elderly and disabled, who can drive the car using only their voices., who can drive the car using only their voices.

## VI. METHODOLOGY

Step 1: Voice commands are been considered to control the vehicle.

Step 2: If the command is Forward then right and left wheel moves clockwise.

Step 3: If the command is Left then right wheel moves clockwise.

Step 4: If the command is Right then left wheel moves clockwise.

Step 5: If the command is Backward then right and left wheel moves anti-clockwise.

Step 6: If the command is Stop then right and left wheel stops moving immediately.

Step 7: Ultrasonic sensors are been considered to stop the vehicle when some object or vehicle interrupts between the range of lesser than 70 and greater than 40.

The below **figure 14** shows the values of ultrasonic sensors in the serial monitor.



Figure.14 Ultrasonic Sensors in serial monitors

```
1. if (data == "o")
{
  Serial.println("Turning off both motors");
  digitalWrite(motor1, LOW);
  digitalWrite(motor11, LOW);
  digitalWrite(motor2, LOW);
  digitalWrite(motor22, LOW);
  analogWrite(motor1speed, 0);
  analogWrite(motor2speed, 0);
}
```

```

}
2. if (data == "f")
{
  Serial.println("Turning off motor 5, Turning on motor 7 ");
  digitalWrite(motor1, HIGH);
  digitalWrite(motor11, LOW);
  digitalWrite(motor2, high);
  digitalWrite(motor22, low);
  analogWrite(motor1speed, 150);
  analogWrite(motor2speed, 150);
}
3. if (data == "b")
{
  Serial.println("Turning on motor 5, Turning off motor 7");
  digitalWrite(motor1, LOW);
  digitalWrite(motor11, HIGH);
  digitalWrite(motor2, low);
  digitalWrite(motor22, high);
  analogWrite(motor1speed, 0);
  analogWrite(motor2speed, 0);
}
4. if (data == "r")
{
  Serial.println("Turning on both motors");
  // Stopping inside motor
  digitalWrite(motor1, LOW);
  digitalWrite(motor11, LOW);
  digitalWrite(motor2, high);
  digitalWrite(motor22, low);
  delay(500);
  stop();
}
5. if (data == "l")
{
  Serial.println("Turning on both motors");
  digitalWrite(motor1, HIGH);
  digitalWrite(motor11, LOW);
  digitalWrite(motor2, LOW);
  digitalWrite(motor22, LOW);
  delay(500);
  stop();
}
6. stop()
{
  digitalWrite(motor1, LOW);
  digitalWrite(motor11, LOW);
  digitalWrite(motor2, LOW);
  digitalWrite(motor22, LOW);
  analogWrite(motor1speed, 0);
  analogWrite(motor2speed, 0);
}
7. if (dist < 70 && dist > 40) {
  digitalWrite(motor1, LOW);
  digitalWrite(motor11, LOW);
}
```





```
digitalWrite(motor2, LOW);  
digitalWrite(motor22, LOW);  
analogWrite(motor1speed, 0);  
analogWrite(motor2speed, 0);  
}
```

The below **figure 15** shows the final implementation of the voice control car.

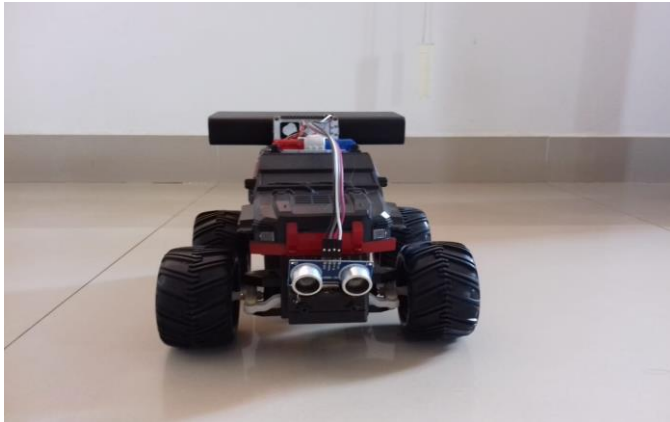


Figure.15 Final implementation of the model

## VII. RESULTS

It's a **voice-controlled robot** vehicle constructed with the NODEMCU esp8266 Controller. The vehicle is powered by various commands such as forward, backward, left, and right. It is essentially linked to a Wi-Fi network. The Android smartphone programme provides the instructions. The car is powered by an android application that connects to a wi-fi network. The **Android application** allows the wi-fi robot car to be quickly relocated from one location to another. Using wi-fi network technology, we can programme the car to perform different tasks. The signals left and right turn the car's wheel left and right, respectively; the inputs left forward and right forward drive the car in the left forward and right forward ways, respectively; and the command left backward pushes the car in the left backward direction.

The robot car is controlled by a **NodeMCU esp8266 controller**, and commands are sent through a wi-fi network to an android application on a smartphone. A built-in wi-fi interface on the NodeMCU esp8266 links the robot to the devices. The Android app will use wi-fi to connect directly to the vehicle, enabling it to drive in the right direction, like forward, reverse, left, right, and stop. The project's main goal is to steer the robotic vehicle into the desired location. In addition, the project's main goal is to use voice control to control the robot. Human-Robot Interaction is now possible. The purpose

of a voice-controlled Robot is to listen to and respond to the user's commands.

## VIII. CONCLUSION

The car is operated by voice in this project of voice control. A mobile robot that can be managed by voice commands is known as a robot vehicle. The speech recognition programme on an Android phone will recognise voice commands such as 'Forward,' 'Stop,' 'Left,' 'Right,' and 'Back,' among others. The operating principle of the robotic car is based on information sent by phone to the robot. The results demonstrate that using only one's voice as a method of control, a user can learn to influence real-world objects reliably. The proposed findings show that voice-controlled robotics would be effective in the future. This device could be used for a wide range of tasks. In the future, systems such as washing machines, microwave ovens, and other home appliances will primarily be voice-controlled. In this situation, the study would be able to effectively meet the current need.

There's really not much space for the robot because it is so small. We can connect the robot from those few metres away because the link between both the robot and the host PC is Wi-Fi. Since we're using a cell phone that all of us have, the platform's cost is minimal. The Internet of Things (IoT) is a rapidly growing industry. For this place, there is only one way forward. In the field of Internet of Things, autonomous robots are a hot topic. Robotics and automation greatly increase efficiency while also reducing resource waste. They are a simple human substitute that poses little risk of death in dangerous environments (for example, factories). While currently limited to a few voice commands, we hope to extend this using Machine Learning and Natural Language Processing.

Android Studio was created with the aim of making testing as easy as possible. With a few clicks, you could build a JUnit test that executes on the local JVM or a designed to operate test that runs on a server. Of course, you can improve your configuration options by using a prototype system in your local test cases to evaluate Android API calls. The ESP8266-01 adds Wi-Fi networking and allows me to monitor them remotely using standard technology and cell phone apps. It's a lightweight, low-cost Wi-Fi transceiver that can be used in a variety of Arduino, Raspberry Pi, and other applications-based projects.

We create a remote-control application for the Android operating system, which is used on smart phones. To communicate with the robot, the software connected through Wi-Fi. Everyone's most basic requirement is wireless control. Wi-Fi modules are used in robots that can be operated through a wireless network. The Android app Node MCU control will issue commands to the car via Wi-Fi, enabling it to drive in the preferred manner, such as forward, reverse, left, right, and stop. In our smart phone, we build a remote-control application. To communicate with the robot, the software uses Wi-Fi. Wireless control is one of the most basic requirements for anyone. Wi-Fi



modules are used in wireless network-controlled robots. The NodeMCU control android application can send commands to the car via Wi-Fi, allowing it to travel in the desired direction, such as forward, reverse, turning left, turning right, and stopping.

#### IX. FUTURE ENHANCEMENT

The Voice Controlled Robot is a prototype that uses a voice activated - Arduino Uno microcontroller board to show the voice control system that can be implemented for a wheelchair. The app requires Bluetooth to transmit these voice signals to the Arduino as a stream of data. The overall project voice-controlled robot vehicle will be used for a multitude of reasons, namely military, surveillance, and human use. It is a wireless robot vehicle that can be controlled by a human voice.

Another advancement is the inclusion of a DHT sensor to sense temperature and relative humidity. Its capacity makes it to reach and enter areas where people are unable, enabling us to detect hot and humid areas (farms where crops grown so large that ground below is hardly visible). As an early and apparent addition, a camera could be attached to the project to lead to better observation from remote locations. However, in Node MCU boards, the issue of small pins arises. We don't have to look too far. However, this comes at the expense of increased difficulty. Nothing in nature is free, as is the case for most things in life. After this is all said and achieved, we should all agree that automated driving and robotics can serve as part of an overall strategy into the future of IoT in the future seasons. This experimental was limited to a short ZigBee device (range 100 meters) linked to the car over a great distance via long-range modules. Sleep and wake-up schedules can be followed to boost energy levels. Image processing can be used in the car to become conscious of the shade and obstacles. Servo motors may be used to provide additional proper service. A robot's focus could be monitored by an automated targeting system.

This project's spectrum has been narrowed to a short-range wi-fi system. Long-range modules and other network equipment will be used to connect with the robot over vast distances. Image processing can be used in the robot to detect colour and artefacts. A thermal camera can be equipped to detect energy created by bodies, which can be useful for defence equipment like tracking attackers on the ground. An Automatic Targeting System can be used by the robot to maintain count of the target. The prototype can be used for residential security and defence purposes by extending the range and incorporating cameras, allowing commands to be given to the robot against risk. The robot comes in handy in situations where humans would be unable to reach but individual voices can be detected. For example, in a fire or in a highly contaminated environment. It is a significant stage in the development of Humanoid robots. Protection devices based on speech and voice recognition. Surveillance and reconnaissance are possible applications for the robot.

#### X. REFERENCE

- [1] R. Pahuja et.al "Android Mobile Phone Controlled Bluetooth Robot," (May 2014).
- [2] M Saravanan et.al developed "Arduino Based Voice Controlled Robot Vehicle" (October 2020).
- [3] H. Jagadish Kumar et.al wrote "Voice Controlled Car using Arduino and Bluetooth Module" (December 2019).
- [4] Aditya Chaudhry et.al implemented "Arduino based voice-controlled Robot" (June 2019).
- [5] Parichart Leechor et.al wrote "Operation of a Radio-Controlled Car by Voice Commands" (March 2010).
- [6] Vipul Mehta et.al gave "Robot Controlled Car Using Wi-Fi Module" (May 2016).
- [7] Mrumal.K.Pathak et.al wrote "Robot Control Design Using Android Smartphone" (February 2015).
- [8] Shubh Srivastava et.al contributed "Voice controlled robot car using Arduino" (May 2020).
- [9] R. Piyare et.al developed "Bluetooth based home automation system using Android phones," (April 2011).
- [10] Ahmed et.al "The development of isolated words pashto automatic speech recognition system." Automation and Computing (ICAC), IEEE Press, pp: 1-4, (September 2014)
- [11] J. Potts et.al "Exploiting bluetooth on android mobile devices for home security application", proceedings of Southeastcon, Orlando, Florida, USA, (February 2012)
- [12] Luo Zhizeng et.al "Speech recognition and its application in voice-based robot control system, Intelligent Mechatronics and Automation," Proceedings, (March 2004).
- [13] Byoung-Kyun Shim et.al "A study on real-time control of mobile robot with based on voice command, Control Automation and Systems (ICCAS)," 11th International Conference on Publication Year, pp. 2011–2011, (March 2011).
- [14] Gyula Mester et.al "Motion Control of Wheeled Mobile Robot" (April 2006)
- [15] J. C. Basilio et.al" Design of PI and PID Controllers with Transient Performance Specification" (October 2002).
- [16] Xiao Lu et.al "Robot Control Design Based on Smartphone" IEEE, 978-1-4673-1382, pp-2820-2823, (June 2013).
- [17] Arpit Sharma et.al "Android phone-controlled robot using Bluetooth" IJEEE, Vol.7, pp-443-448, (November 2014)
- [18] Peter X. Liu et.al "Voice Based Robot Control", The Proceeding of the 2005 IEEE International Conference on Information Acquisition, pp. 543-547, (January 2005).
- [19] Tetsuya Takiguchi et.al "Human-Robot Interface Using System Request Utterance Detection Used on Acoustic Features", The 2008 International Conference on Multimedia and Ubiquitous Engineering, pp. 304-309, (April 2008).
- [20] J. Potts et.al wrote "Exploiting bluetooth on android mobile mobile devices for home security application" (October 2014).
- [21] Vito M. Guardi et.al "Design of a Bluetooth Enabled Android Application for a Microcontroller Driven Robot" (May 2014).



[22] Ranjith Kumar Goud et.al “Android Based Robot Implementation for Pick and Retain of Objects” (October 2014).

[23] M.Selvam et.al “Smart phone based robotic control for surveillance applications” (January 2014).

[24] Kishan Raj KC et.al “Controlling a Robot using Android Interface and Voice” (January 20120).

[25] Arpit Sharma et.al “Android phone-controlled robot using Bluetooth” IJEEE, Vol.7, pp-443-448, (November 2014).