**SYMBIOSIS UNIVERSITY OF APPLIED SCIENCES**

**INDORE**



PROJECT

ON

“**Wi-Fi Controlled Car**”

Submitted to “Symbiosis University of Applied Sciences, Indore

BACHELOR OF TECHNOLOGY

IN

SCHOOL OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

**Submitted To:** **Submitted By:**

Mr. Ishwarlal Rathod Yaman Kulkarni (2020BTCS033) Assistant Proffesor Yash Khilavdiya (2020BTCS041)

Tejas Bhati (2020BTCS042)

**SYMBIOSIS UNIVERSITY OF APPLIED SCIENCES**

**INDORE**

# CERTIFICATE

This is to certify that the Internship report entitled “**Wi-Fi Controlled Car**”, submitted by Yaman Kulkarni, Yash Khilavdiya, Tejas Bhati students of the Third year towards partial fulfillment of the degree of Bachelor of Technology in the School of Computer Science and Information Technology in the academic year 2022-2023. Symbiosis University of Applied Sciences, Indore (M.P.) is in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology and is a bonafide record of the work carried by Yaman Kulkarni, Tejas Bhati, Yash Khilavdiya, during the academic year 2022-2023.

Place: Indore (M.P) Date:

INTERNAL EXAMINER EXTERNAL EXAMINER

**CONTENTS**

[CERTIFICATE 2](#_Toc135084218)

[Introduction: 4](#_Toc135084219)

[How does it work: 4](#_Toc135084220)

[Circuit Diagram: 5](#_Toc135084221)

[Required Hardware Component: 5](#_Toc135084223)

[Hardware Components Usage: 6](#_Toc135084225)

[Required Software Component: 8](#_Toc135084226)

[Arduino IDE: 8](#_Toc135084227)

[Blynk APK: 8](#_Toc135084228)

[Code: 8](#_Toc135084229)

[Output: 12](#_Toc135084230)

[Conslusion: 15](#_Toc135084233)

[Challenges Faced: 15](#_Toc135084234)

[References: 15](#_Toc135084235)

# Introduction:

In this project we have build a wifi controlled car using NodeMCU and Blynk App to establish a wireless connection and enable remote control through a WiFi network. NodeMCU is an open-source firmware and development kit based on the ESP8266 microcontroller, which provides WiFi connectivity and a range of features that make it an ideal platform for building WiFi controlled cars.

## How does it work:

The concept of Wi-Fi controlled succeeds majorly using NodeMCU and BlynkApp.

Following is the brief description of working of our Wi-Fi controlled car: -

1. First, when we connect our whole circuit with the Power Supply, then the NodeMCU creates a server with the given SSID and the Password. Now we connect our NodeMCU with local Wi-Fi network.
2. Then NodeMCU runs a program that communicates with the Blynk server using the Blynk library.
3. Then Blynk APK is installed on a smartphone or tablet, which also connects to the local Wi-Fi network.
4. Now we can use Blynk APK to control our car using an application which consist of a joystick and slider. Joystick sends the data to the NodeMCU which controls the movement of the car such as to move forward, backward and change direction etc.

## Circuit Diagram:

# 

# Required Hardware Component:

# 

**NodeMCU- ESP8266 Motor Driver - L298N**

**9V Battery Universal Swivel Castor Wheels**

 **Wheels Single Shaft BO motor**

**Jumper Wires**

# Hardware Components Usage:

|  |  |  |
| --- | --- | --- |
| **Item Name** | **Photo** | **Description** |
| **Universal Swivel Castor Wheels** | **https://cdn.shopify.com/s/files/1/0559/1970/6265/products/137_9067b6bb-ad56-40ee-a07b-8496480fdc51_1024x1024@2x.jpg?v=1670582294** | It acts as front wheel for our car which help us in changing the direction of the car. |
| **NodeMCU ESP8266** | **Buy ESP8266 NodeMCU CP2102 Board at Lowest Price In India | Robu.in** | It is a microcontroller board with inbuild Wifi build which is a major part of our project which helps in connecting our car (NodeMCU) with mobile phone through Wi-Fi Connectivity. |
| **Jumper Wires** | **ApTechDeals Jumper Wires Male to Male, Male to Female, Female to Female/breadboard  jumper wires (10+10+10) : Amazon.in: Industrial & Scientific** | These are used for making connection between various components used in our project. |
| **L298N 2A Based Motor Driver Module** | **C:\Users\tejas\Downloads\WhatsApp Image 2023-04-04 at 09.33.48.jpeg** | For Interfacing motors with NodeMCU. Since Motors requires an high voltage to operate but NodeMCU could not provide high power supply so we supply it though battery and for interfacing it we need Motor Driver. |
| **Durable Rubber Tire Yellow wheel 45mm for BO DC Motor** | https://cdn.shopify.com/s/files/1/0559/1970/6265/products/111_4e83be32-e117-4d0a-8449-e1f6edf048de_1024x1024@2x.jpg?v=1670581242 | For movement of our Car. Direction of Movement of Tyres decide the direction in our car will move. Such as Forward, Backward, Right and Left movement. |
| **Battery Holder** |  | It is used for supply power to all the components or circuit of our project. |
| **Single Shaft BO motor**  **3v-6v** | Generic: Single Side Shaft BO Motor | As the functionality of motor is rotation so for moving our car we require it. |

# Required Software Component:

## Arduino IDE:

We have used Arduino IDE for Code Development and Uploading code by which we were able to test our components and run our final project. We have use libraries for Blynk APK and Esp8266 WiFi Module for building our project. Since we have used NodeMCU microcontroller Board so we also required to add NoduMCU Board Library using Board Manager in Arduino IDE.

## Blynk APK:

On Blynk Cloud we have created an account and build a device which consists of Virtual Pins and can fetch data form the Blynk App and send it to NodeMCU Board.

Using Blynk APK we have developed a mobile application using which we can control our car. Our application consist of joystick and if any movement in joystick occurs then we move our car accordingly.

## Code:

// Including Libraries

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

// Define the Motor Pins

#define ENA D0

#define IN1 D1

#define IN2 D2

#define IN3 D3

#define IN4 D4

#define ENB D5

#define BlinkLED D6

// Variables for the Blynk widget values

int x = 50;

int y = 50;

int Speed = 255;

int speed\_Coeff = 3;

// Authorization Keys and Info for Connecting wifi module with mobile phone

char auth[] = "Cfr2SS\_lR6WZFnTDwrLBbFjTrOLJMxP1"; //Enter your Blynk auth token

char ssid[] = "Yash Khilavdiya Galaxy M33"; //Enter your WIFI name

char pass[] = "jsk-1941"; //Enter your WIFI passowrd

void setup() {

Serial.begin(9600);

//Set the motor pins as output pins

pinMode(ENA, OUTPUT);

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

pinMode(IN3, OUTPUT);

pinMode(IN4, OUTPUT);

pinMode(ENB, OUTPUT);

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

// Initialize the Blynk library

Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);

}

// Get the joystick values

BLYNK\_WRITE(V0) {

x = param.asInt();

Serial.println("X : " + x);

}

//// Get the joystick values

BLYNK\_WRITE(V1) {

y = param.asInt();

Serial.print("Y : " + y);

}

////Get the slider values

BLYNK\_WRITE(V2) {

Speed = param.asInt();

Serial.println("Speed : " + Speed);

}

// Check these values using the IF condition

void smartcar() {

if (y > 70) {

carForward();

Serial.println("carForward");

}

else if (y>55 & x>55){

carForwardLeft();

Serial.println("Moving Forward in Left Direction");

}

else if (y>55 & x<15){

carForwardRight();

Serial.println("Moving Forward in Right Direction");

}

else if (y<15 & x>55){

carBackwardLeft();

Serial.println("Moving Backward in Left Direction");

}

else if (y<15 & x<15){

carBackwardRight();

Serial.println("Moving Backward in Right Direction");

}

else if (y < 30) {

carBackward();

Serial.println("carBackward");

} else if (x < 30) {

carLeft();

Serial.println("carLeft");

} else if (x > 70) {

carRight();

Serial.println("carRight");

} else if (x < 70 && x > 30 && y < 70 && y > 30) {

carStop();

Serial.println("carstop");

}

}

void loop() {

Blynk.run();// Run the blynk function

smartcar();// Call the main function

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*Motor movement functions\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void carForward() {

// Turning off LED when car is not moving backwards

digitalWrite(BlinkLED, LOW);

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

void carBackward() {

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

// Blinking LED when car is moving backwards

digitalWrite(BlinkLED, HIGH);

}

void carLeft() {

// Turning off LED when car is not moving backwards

digitalWrite(BlinkLED, LOW);

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

void carRight() {

// Turning off LED when car is not moving backwards

digitalWrite(BlinkLED, LOW);

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

}

void carForwardRight(){

// Turning off LED when car is not moving backwards

digitalWrite(BlinkLED, LOW);

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

analogWrite(ENA, Speed/speed\_Coeff);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

analogWrite(ENB, Speed);

}

void carForwardLeft(){

// Turning off LED when car is not moving backwards

digitalWrite(BlinkLED, LOW);

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

analogWrite(ENA, Speed);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

analogWrite(ENB, Speed/speed\_Coeff);

}

void carBackwardRight(){

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

analogWrite(ENA, Speed/speed\_Coeff);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

analogWrite(ENB, Speed);

// Blinking LED when car is moving backwards

digitalWrite(BlinkLED, HIGH);

}

void carBackwardLeft(){

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

analogWrite(ENA, Speed);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

analogWrite(ENB, Speed/speed\_Coeff);

// Blinking LED when car is moving backwards

digitalWrite(BlinkLED, HIGH);

}

void carStop() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, LOW);

// Turning off LED when car is stopped

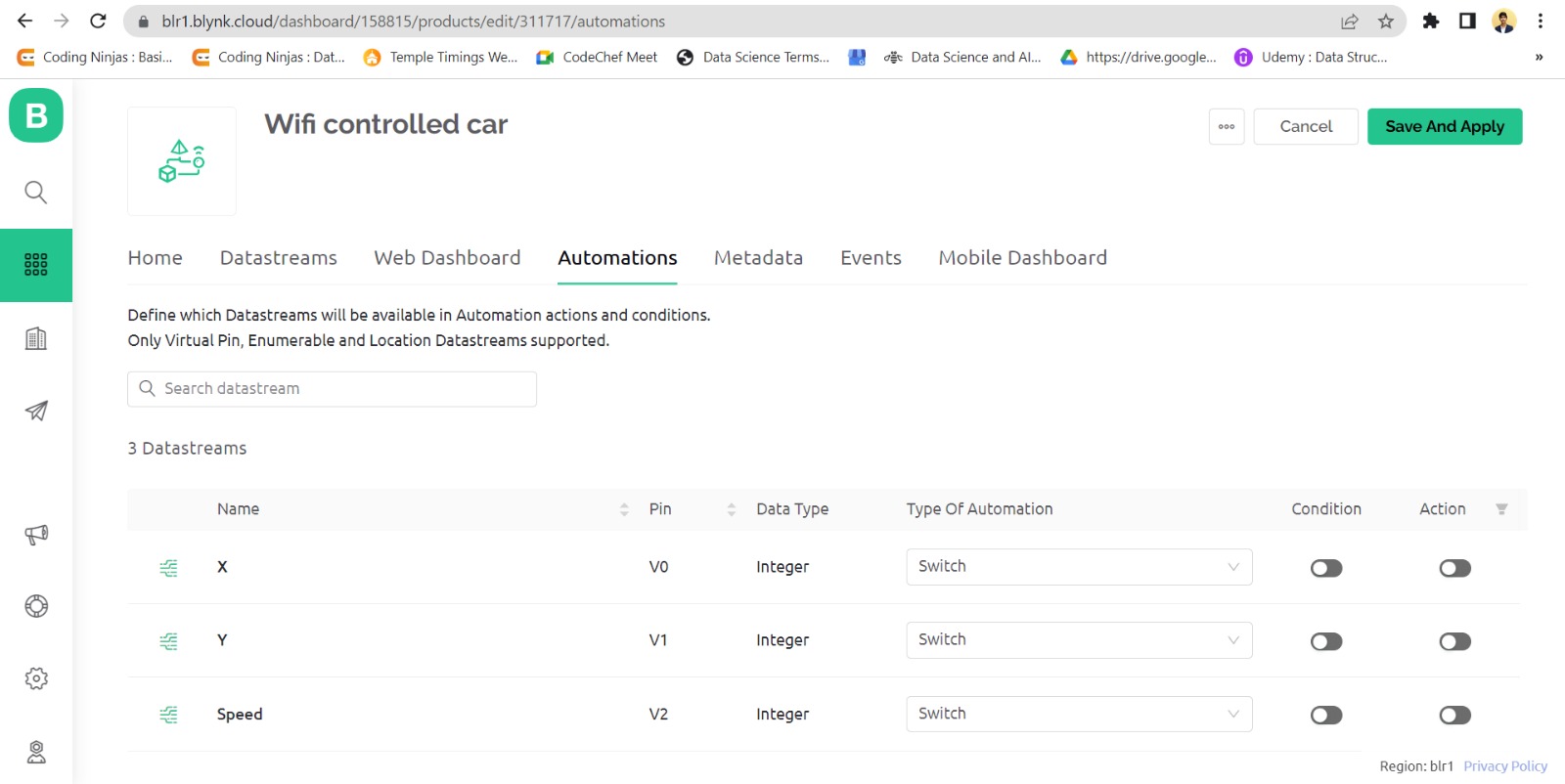
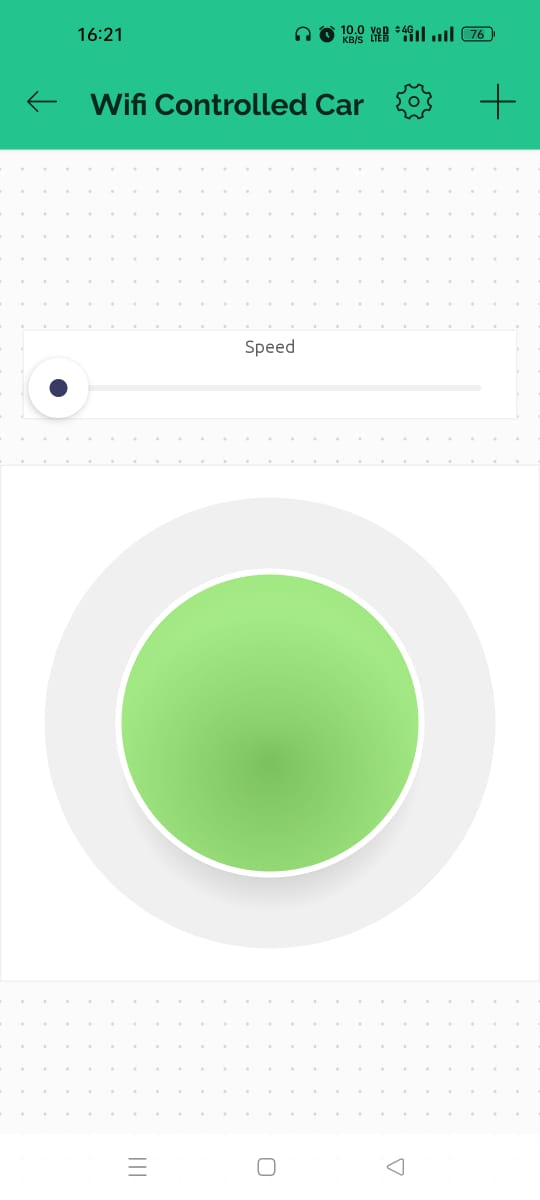
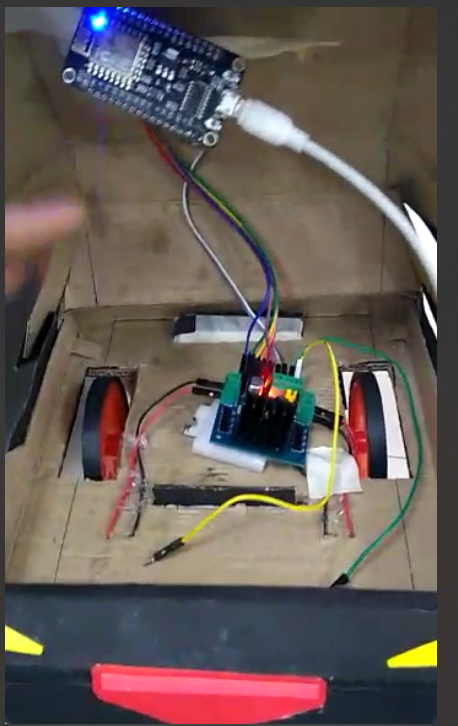
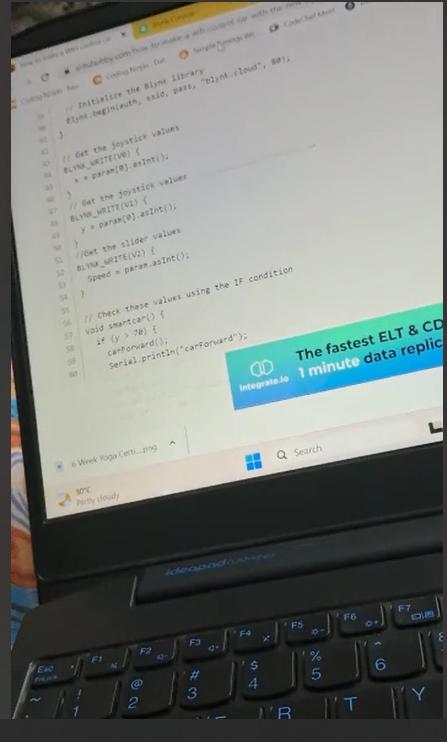
digitalWrite(BlinkLED, LOW);

}

# Output:

# C:\Users\tejas\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\645E6BFDD05D1A69C5E47B20F0A91D46\WhatsApp Image 2023-05-15 at 21.55.23.jpgC:\Users\tejas\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\946E3ECE1FC8B24BD656449D88ECA941\WhatsApp Image 2023-05-15 at 21.54.57.jpg C:\Users\tejas\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\649646790E6BCAEEBF1E1DB287542B77\WhatsApp Image 2023-04-27 at 23.15.59.jpg C:\Users\tejas\AppData\Local\Packages\5319275A.WhatsAppDesktop_cv1g1gvanyjgm\TempState\9D4A91B041C060C2FBB3F7E3A806DBA6\WhatsApp Image 2023-04-27 at 23.15.59.jpg

# Blynk: a low-code IoT software platform for businesses and developersArduino Integrated Development Environment (IDE) v1 | Arduino DocumentationC:\Users\tejas\Music\WhatsApp Image 2023-04-27 at 23.15.49.jpg



# Conslusion:

It was really an exciting and fun project and our team enjoyed building it. It was really a learning journey by building our project as well as from our mistakes. So Wi-Fi controlled car can be used in building toy cars for kids as well as for visiting places where humans can’t visit.

# Challenges Faced:

Major problem in our project was that 9 Volt battery was not able to supply sufficient power to motor driver and NodeMCU. So, we used multiple batteries to generate around 20 Volts of power supply, but even after after that our circuit was not working properly so we decided to give seperate power supply to NodeMCU and Motor Driver but then also it was not working fine. But with laptop or power bank power supply it was working properly. So we decided to use power bank for our project.

# References:

**Online Articles: -**

<https://srituhobby.com/how-to-make-a-wifi-control-car-with-the-new-blynk-app-step-by-step/> <https://circuitbest.com/arduino-smartphone-controlled-wifi-car-using-nodemcu-esp8266/> <https://www.instructables.com/WiFi-Car-Using-NodeMCU/>

<https://www.hackster.io/yasirutishan/wifi-control-car-arduino-concept-internet-of-things-26a041> <https://aarnavjindal.medium.com/mobile-controlled-car-node-mcu-6f0e3efe82eb>

**You Tube Videos: -**

<https://youtu.be/fr9i0CBDtms>

<https://www.youtube.com/watch?v=zJnDbdefeCA>

<https://www.youtube.com/watch?v=OIaSYfVZyfg>