MOBILE ROBOTIC ARM WITH APP INTERFACE USING MICROCONTROLLERS

Kartikeya Mandhar[1], Abhinav Thapa[2], Naman Aggarwal[3]

Department of Communication Engineering
School of Electronics Engineering
Vellore Institute of Technology, Vellore, Tamil Nadu, India

Abstract - *This paper presents the working of Mobile Robotic Arm with App Interface using Microcontrollers. It constitutes the interaction between different software and hardware components as it's backbone. The microcontroller used for the robotic car and sensor reading is the NodeMCU, the microcontroller used for the robotic arm is an Arduino UNO board. There are two apps developed using python, one for the car and for the arm. The apps interact using the principle of serial communication as this concept provides a minimal delay between the app and the respective hardware response. Two IR sensors to determine the picked and reached status and one Ultrasonic sensor determine object distance. All the hardware programming is done on Arduino IDE, apps have been made on SPYDER IDE, and the website has been made on Microsoft Visual Studio. Information from the sensors is sent to firebase database, from firebase we extract the data onto our website using JavaScript API's and the website has been made using HTML, CSS, JavaScript, and Bootstrap.*

Github Link - https://github.com/kartikeyamandhar/Mobile- Robotic-Arm-with-App-and-Web-Interface

Key Words: Robotic, Microcontrollers, Python, Serial Communication, Database, Integration.

1. INTRODUCTION

Today innovation is creating similar way in accordance with quickly expanding human necessities. The work done to address these issues makes life simpler consistently, and these studies are packed in robotic arm studies [1]. The robot arm works with an external client or by performing foreordained orders [2-4]. Many people cannot move from one place to another place for their limitation but they have needed to move for collect something like mug, jog, and so on. For that they require getting help from other persons [5-7]. When they use this type of robot, they can solve their problem easily without help other person for its easy operation system. The design and implementation of the whole arm along with the mobility control has been exceedingly interesting, as it makes full use of various Electronics and Software fields such as, sensors, microcontrollers, embedded, python development, database management and web development. This project was chosen because of the amalgamation it provides between our subjects and the industrial presence of the same. It has massive applications in industries such as medicine, construction, automotive, etc.

2. METHODOLOGY

The steps to develop the whole prototype are listed below-

1. Designing of the Robotic Car would be done using NodeMCU, one ultrasonic sensor and one IR sensor.

2. The app to control the car, would be made using Python, and interact with the car on the basis of Serial Communication.

3. Designing of the Robotic Arm would be done using Arduino UNO, Four servo motors and one IR sensor.

4. The app to control the arm, would be made using Python, and interact with the car on the basis of Serial Communication.

5. The app to control the car would have 5 buttons forward, leftward, backward rightward and stop, to respectively control the 4 wheels of the car.

6. The app to control the arm has 4 sliders to control by how much angle would you move the respective arms.

7. The data that we would send via the apps would be sent as bytes to a serial port, from where our microcontroller would interpret them, and make a decision that would lead to physical output.

8. IR sensor on the claw of the arm would determine the status of the object being picked or not.

9. IR sensor on the car would determine whether we have reached the object or not

10. Ultrasonic sensor would determine how far the object is in centimeters.

11. All the sensor data would then be realized, and sent to firebase in real-time. As and when data is being appended to firebase, the website developed using HTML, CSS, Bootstrap and JavaScript would extract the data, and update it in a

tabular, user-friendly format.

12. The results would be stored in three sections. Distance – stores the result from the ultrasonic sensor, picked – stores the result form IR sensor on the car, Status - stores the resultform IR sensor on the car.

3. DESIGN AND IMPLEMENTATION

3.1 ROBOTIC ARM

For developing the robotic arm, Arduino UNO microcontroller has been used along with 4 servo motors (figure 1). The reason Arduino board has been used to develop the arm is its ability to send data with PWM, as servo motors can only read input which has pulse width. An Arduino board provides the functionality of having 6 PWM pins. A servo motor is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback of the current position.

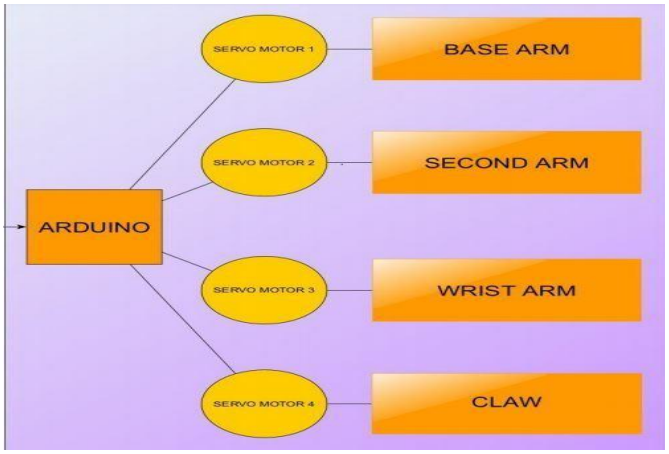

Fig-1 Hardware Interaction for Robotic Arm

Four sliders are present to control the angle of the 4 servo motors (figure 2). The range of changing the angle is from 0degree to 180 degrees for all 4 sliders. When we move a slider from 0 to a particular angle value, the data is sent to the port, which will further be read by the Arduino UNO. The Arduino code is written in such a way that it will read the data incoming in string format, check its last three characters and then check which corresponding servo motor to write the data to. Since the App provides the functionality of moving 4 sliders in sequence, the data that is being sent would be either a 2-digit or a 3-digit number (for example 89 degrees or 176 degrees) However, when we move the sliders, we have no way of differentiating that the integer angle data being sent to the port, came from which slider, because what all the functions are doing, is sending data to the port in integer format, not knowing which slider it belongs to, hence resulting in all the 4 servos moving by the same angle value. In order to troubleshoot this error and make sure that at a time only the servo corresponding to a particular slider is moved, we have added 3-Digit integer values to the angle data (figure 3).

Fig-2 App Interface for Arm control

3.2 ROBOTIC CAR

When a button is pressed on the app (figure 6) to control the car, a byte is sent to the port.

1. If forward is pressed, then the byte is 'F'
2. If backward is pressed, then the byte is 'B'
3. If leftward is pressed, then the byte is 'L'
4. If rightward is pressed, then the byte is 'R'
5. If stop is pressed, then the byte is 'S'

Any byte could have been sent to the port however simple bytes have been so as to enhance the readability of the code. Now, byte a byte is ready on a port on our PC, and the NodeMCU is ready to interpret what it is, code written in the Arduino IDE would de-code what the byte is and run the respective function to be sent to the motor to run the required wheels for locomotion. In the Arduino IDE the code has been written for the NodeMCU (figure 5), which would read the byte at the port, after which it will recognize which byte it is and what function to send to the motor driver (figure 4). The motor driver would then make the necessary motors move.

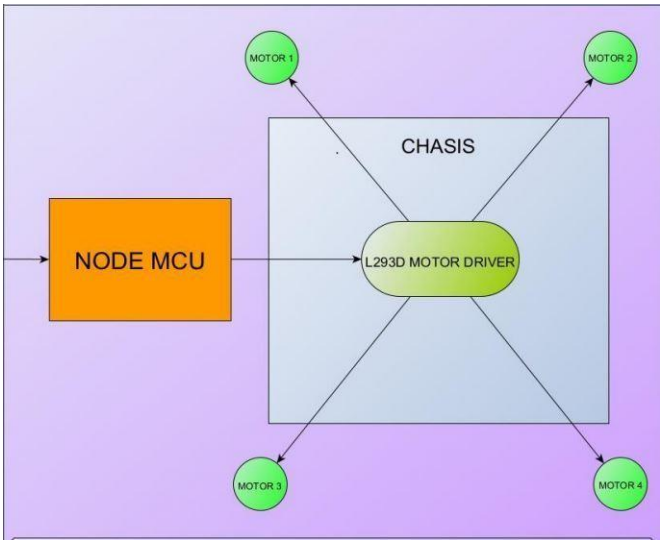

Fig-5 Hardware Interaction for Robotic Car

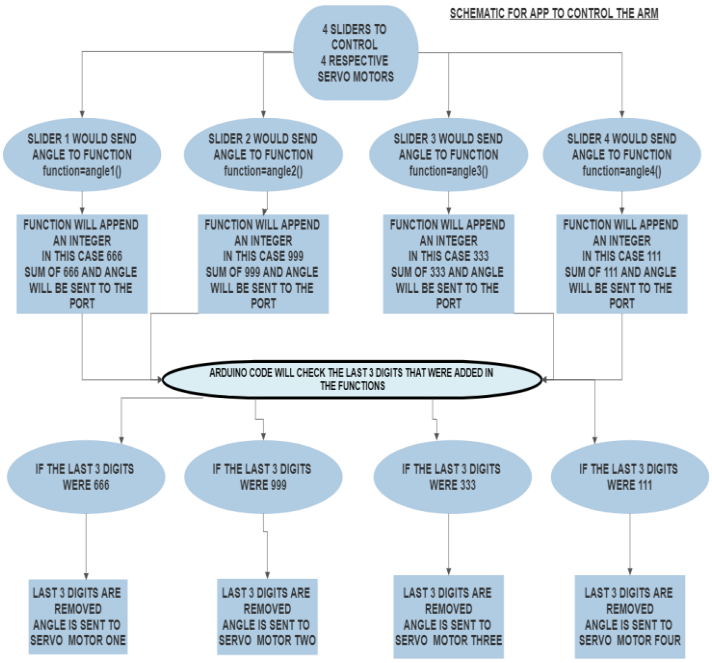

Fig-3 Flowchart description for app control of robotic arm

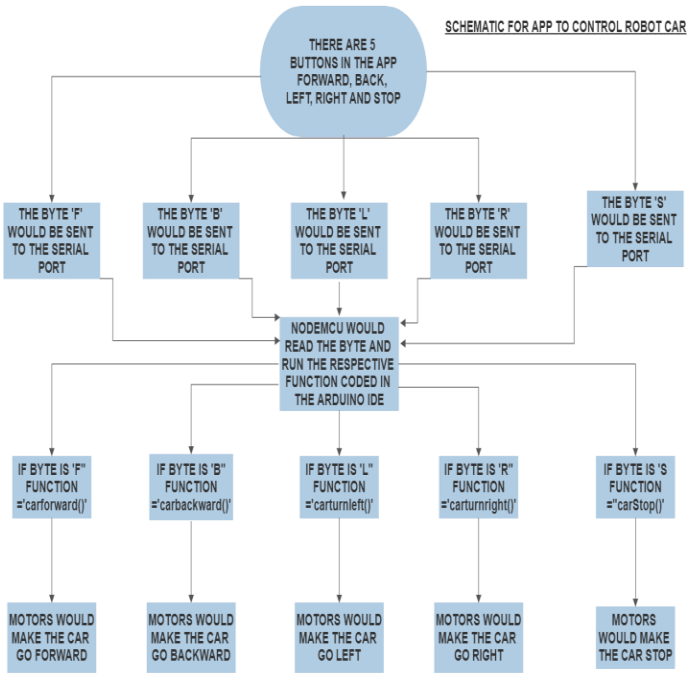

Fig-4 Flowchart description for app control of robotic car

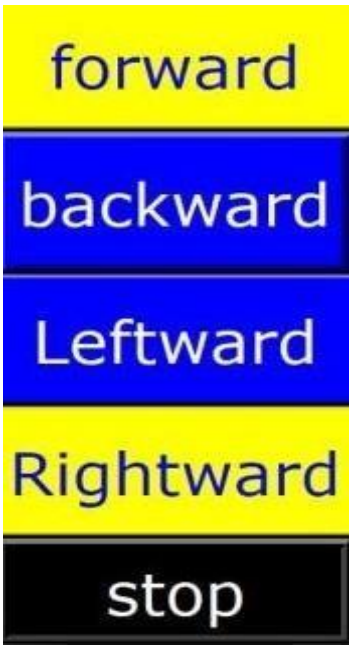

Fig-6 App interface for Car Control

3.3 FRONT-END TO DISPLAY PANEL

The communication between the sensors and the webpage could be divided into two parts-

- Sensor-Firebase interaction
- Firebase-Webpage interaction

3.3.1 SENSOR-FIREBASE INTEGRATION

A total of three sensors have been used, 1 Ultrasonic sensor and 1 IR sensor on the robotic car, and 1 IR sensor on the claw of the robotic arm. The NodeMCU has been used for taking the readings from the sensors, as NodeMCU comes with an ESP8266 Wi-Fi module that enables it to interact with other hardware devices or software APIs via the internet.

3.3.2 FIREBASE-WEBPAGE INTEGRATION

Firstly, the basic backbone of the webpage is designed using HTML and CSS (figure 8), then the table was made. After declaring the table, a function is made to select the data referencing the Firebase database by name. Then, API provided by firebase unique to the database is added to the code so that interaction between the website code and firebase can be established. After establishing connection between firebase and the code, as well as selecting the data, addition of the data is done to the table. Styling for the same is done using CSS and Bootstrap.

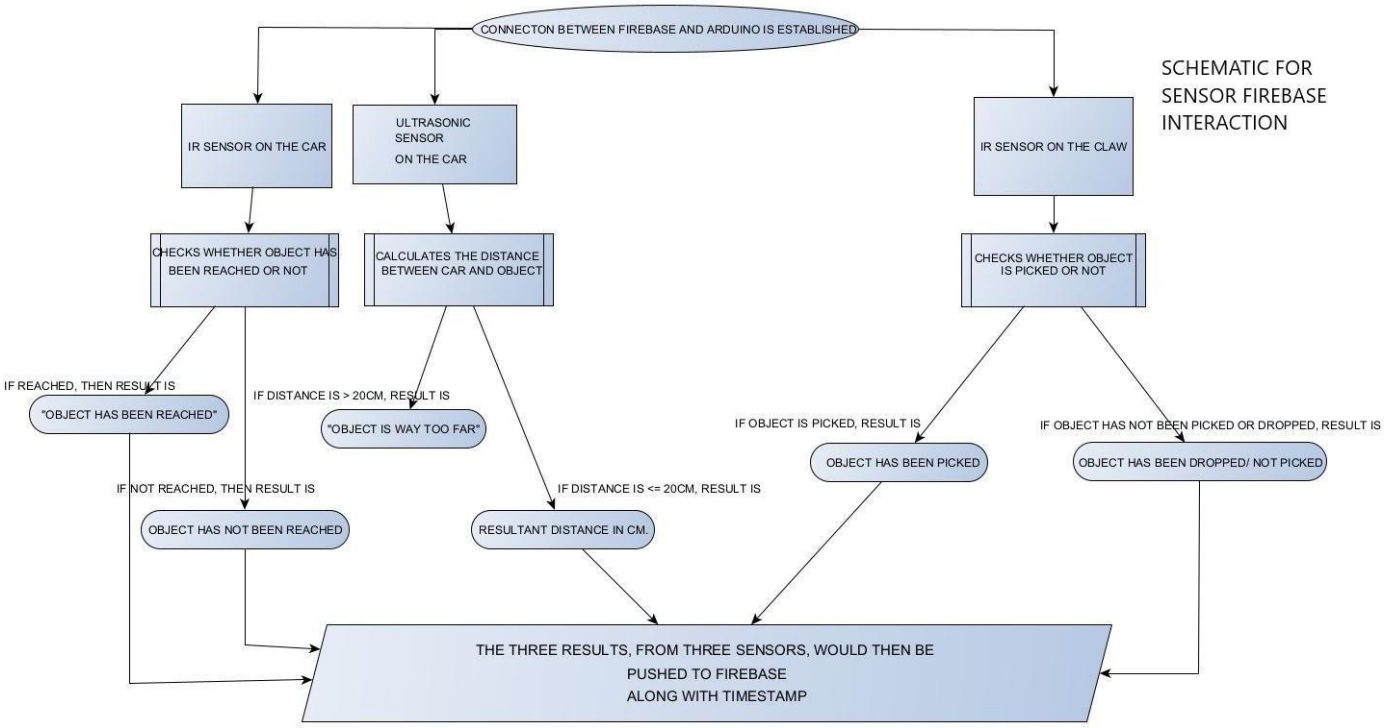

Fig-7 Flowchart description for Sensor-Firebase interaction

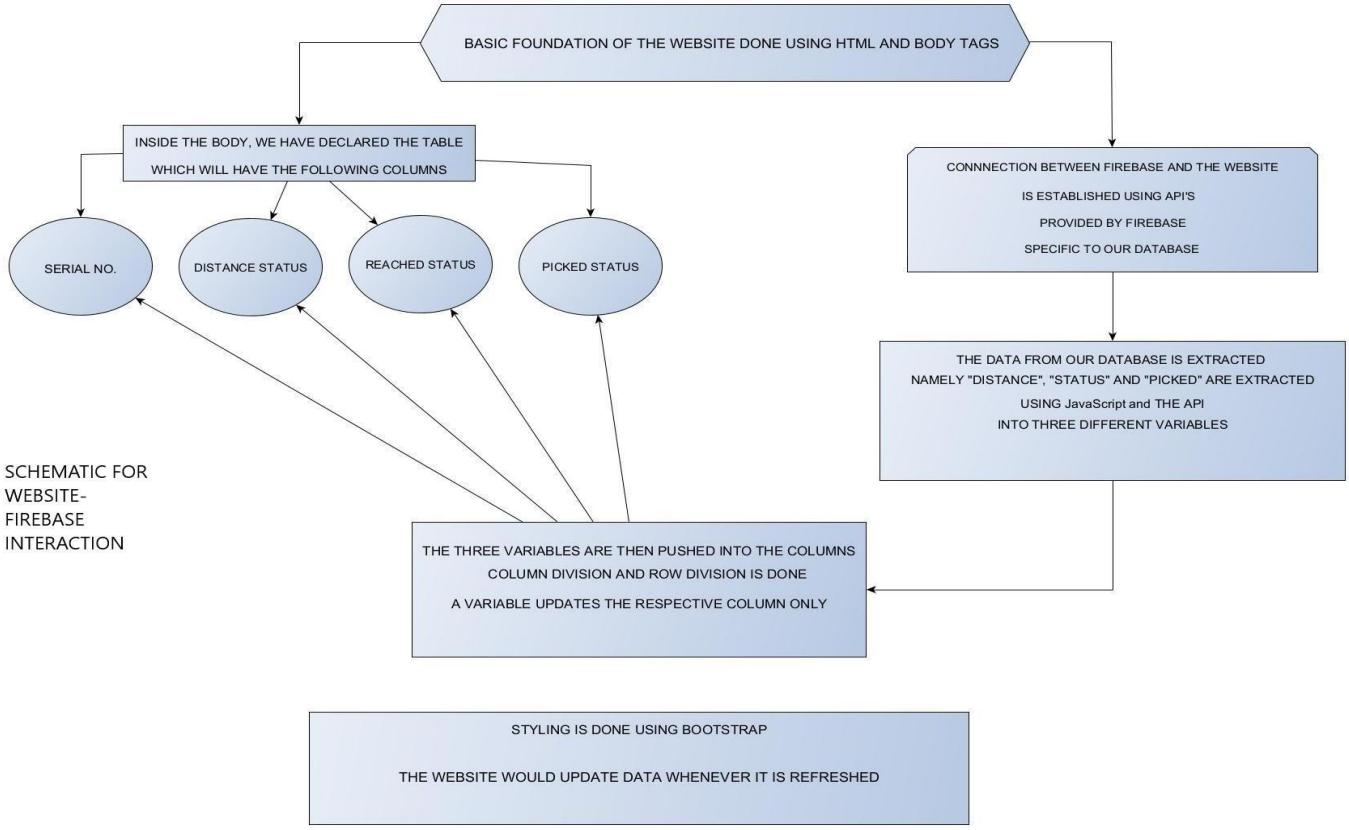

Fig-8 Flowchart description for Firebase-Website Interaction

4. RESULTS AND DISCUSSION

In the resultant database, we have the following – roboarm-345f1-default-rtdb (figure 11) is the name given by firebase to the dataset. As one can see timestamps have helped in segregating the data and sent the three resultant strings inpackets. The strings have been sent in three variables- Distance (stores the result from the ultrasonic sensor), Picked (stores the result form IR sensor on the car) and Status (stores the result form IR sensor on the car) All thesevariables would be extracted by the Webpage. For the IR sensor on the car, if value is low, which means no data is being read on the IR sensor, which means Object has been reached. Therefore, the resultant string" Object has been reached" (figure 7) would be stored in the variable called "value", which will be pushed to firebase. If value is high, meaning the Object has not been reached, hence pushing thestring "Object has not been reached" For the IR sensor on theclaw of the arm, if value is low, which means no data is being read on the IR sensor, which means Object has been picked. Therefore, the result string" Object has been picked" wouldbe stored in the variable called "value", which will be pushed to firebase. If value is high, meaning the Object has not been picked, hence pushing the string "Object has not been dropped/not picked" For the ultrasonic sensor on the car, If the distance is more than or equal to 20cm, the resultant string will be sent as "Object is way too far", if the distance is less than 20cm, the actual reading would be stored as a string, the result would be sent to firebase.

• Distance Status: When distance between the object and the arm is more than 20cm it will show "Object is way too far" (figure 10), if less than 20cm, then it will show the actual numerical reading in cm. Calculated using data from the Ultrasonic Sensor on the robotic car.

• Reached Status: Measured from the IR sensor on the car, will show "Object has been reached" when reached the object, will show "Object has not been reached" otherwise.

• Picked Status: Measured from the IR sensor on the hand of the arm, will show "Object has been picked" when picked the object, and will show "Object has been dropped/not picked" otherwise.

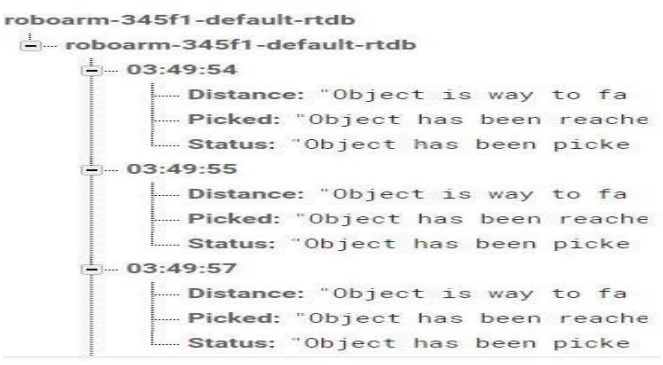

Fig-11 Firebase Database

Serial No.	Distance Status	Reached Status	Picked Status
1	Object is way to far	Object has been reached	Object has been picked
2	Object is way to far	Object has been reached	Object has been picked
3	Object is way to far	Object has been reached	Object has been picked
4	Object is way to far	Object has been reached	Object has been picked
5	Object is way to far	Object has been reached	Object has been picked
6	Object is way to far	Object has not been reached	Object has been dropped/not picked
7	Object is way to far	Object has not been reached	Object has been dropped/not picked
8	Object is way to far	Object has not been reached	Object has been dropped/not picked
9	Object is way to far	Object has not been reached	Object has been dropped/not picked

Fig-10 Website Status Display

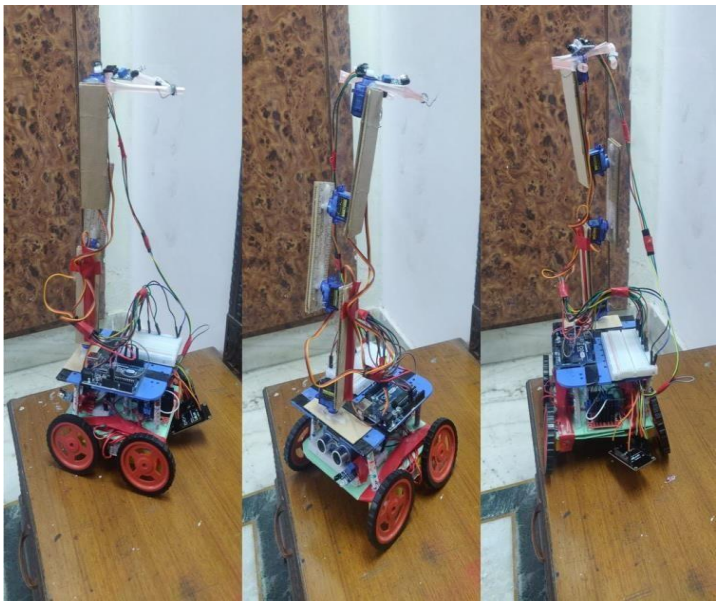

Fig-9 Proposed Mobile Robotic Arm

5.CONCLUSION

The design and implementation of the whole arm along with the mobility control has been exceedingly interesting, as it makes full use of various Electronics and Software fields such as, sensors, microcontrollers, embedded, python development, database management and web development. This project was chosen because of the amalgamation it provides between our subjects and the industrial presence of the same. It has massive applications in industries such as medicine, construction, automotive, etc. It proves to be editable according to various needs as well. For instance, one can add more components without altering the previous design. The aim along with the massive smooth integration of all components, was the ease of use of our whole device. The two app controls, have been developed, keeping in mid the ease of interaction required that will make it effortless for the user to control them. In the website, the results are displayed in a systematic manner and anyone reviewing the

project can easily make sense of what the hardware is doing and what its results are. Excessive troubleshooting and development were done to make all the components work together along with an e easily operable UI. The project is a mix of various technological concepts working without faults to produce results.

6. REFERENCES

[1] Diodata Jr MD, Prasad SM, Klingensmith ME, Damiano Jr RJ (2004) Robotics in surgery. Current Prob in Surg 41(9):748–810

[2] Modeling and Analysis of a 6 DOF Robotic Arm Manipulator Jamshed Iqbal, Raza ul Islam, and Hamza Khan Canadian Journal on Electrical and Electronics EngineeringVol. 3, No. 6, July 2012

[3] Development of a Microcontroller Based Robotic Arm Jegede Olawale, Awodele Oludele, Ajayi Ayodele, and NdongMiko Alejandro Babcock University, Ilisan- Remo, Nigeria 2007 Computer Science and IT Conference

[4] Miyake N (1986) US Patent 4,568,311. "Flexible wrist mechanism." Filed 11 January 1983 and issued 4 February 1986

[5] Design, Analysis and Implementation of a Robotic Arm the Animator.Md. Anisur Rehman, Alimul Haque Khan, Bangladesh University

[6]A Geometric Approach for Robotic Arm Kinematics with Hardware Design, Electrical Design, and Implementation Kurt E. Clothier and Ying Shang Department of Electrical & Computer Engineering, Southern Illinois University Edwardsville

[7]Review on Development of Industrial Robotic Arm Rahul Gautam, Ankush Gedam, Ashish Zade, Ajay Mahawadiwar Abha Gaikwad Patil College of Engineering, Nagpur, Maharashtra, India