A

Major Project

On

ADVANCED SURVEILLANCE ROBOT WITH ROBOTIC ARM CONTROL OVER IOT

(Submitted in partial fulfilment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

A. MEGHANA (187R1A0564)

D. JAYENDRA SAI VARMA (187R1A0574)

CH. ASHRITHA REDDY (187R1A05A8)

B. LALITHA (187R1A0563)

Under the Guidance of

M. SUNITHA (Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956, Kandlakoya (V),Medchal Road, Hyderabad-501401.

2018-22

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**ADVANCED SURVEILLANCE ROBOT WITH ROBOTIC ARM CONTROL OVER IOT**" being submitted by **A.MEGHANA (187R1A0564), D.JAYENDRA SAI VARMA (187R1A0574), CH.ASHRITHA REDDY (187R1A05A8) & B.LALITHA (187R1A0563)** in partial fulfilment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2021-22.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

M. SUNITHA
Assistant Professor
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HoD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on _______________

ACKNOWLEDGMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Mrs.M Sunitha**, Assistant Professor for her exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by her shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) **Mr. A. Uday Kiran, Mr. J. Narasimha Rao, Dr. T. S. Mastan Rao, , Mr. A. Kiran Kumar & Mrs. G. Latha** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju**, Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy**, Director for being cooperative throughout the course of this project. We also express our sincere gratitude to **Sri. Ch. Gopal Reddy**, Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

A. MEGHANA	**(187R1A0564)**
D. JAYENDRA SAI VARMA	**(187R1A0574)**
CH. ASHRITHA REDDY	**(187R1A05A8)**
B. LALITHA	**(187R1A0563)**

ABSTRACT

In the present world everyone is in need of the safety due to increase in crime rate this has led to an increase in importance of surveillance system. Surveillance is the monitoring of behavior activities or other changing information usually of people for the purpose of influencing, managing, directing or protecting them. Project is to design and build a manually controlled surveillance robot. The main purpose of the robot is to be able to roam around in a given environment while transmitting back real time data (video) to the ground station. This real time data can then be used by the controller (human) to move the robot around. This robot is mainly designed for army operations which has robotic arm in front to do pic and place operation and etc which is very useful for ARMY application and INDUSTRIAL applications. We are using raspberry pi a small credit card size computer which will control the robot directions and transmits the live real time video to ground station over IOT with Wi-Fi network for human for reducing his life from risky operations.

LIST OF FIGURES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1.INTRODUCTION

1. INTRODUCTION

1.1 PROJECT PURPOSE

This project has been developed to focus on a robot that is designed solely to serve the purpose of moving or discarding hazardous material where the users can not be present and surveillance of the surrounding area to understand the environment in which the robot will be operated.

1.2 PROJECT SCOPE

The Robotics Institute of America (RIA) defines a robot as a "reprogrammable multifunctional manipulator", which allows materials, parts, tools, or special tools designed to move. "RIA" subdivides the robots into four classes: tools that use objects with manual control, automated devices that use objects with preestablished cycles, programmable and controlled by all robots with a constant path from point to point, and the last type that receives information of the environment. In industry, robotics is a step beyond mechanization. Engineers try to combine robots with mathematical and organizational tools by using various applications to build complex systems for a rapidly growing range of activities.

1.3 PROJECT FEATURES

In the purposed system esp32 is used as controller and to get the information of real-world a camera will be used. This robot and its arm can be controlled using mobile through wifi. This vehicle control system is used will help in reducing accidents that happens on the path. camera is used give information about the environment.

2.SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

SYSTEM ANALYSIS :

System Analysis is the important phase in the system development process. The System is studied to the minute details and analysed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, "what must be done to solve the problem?" The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

2.1 PROBLEM DEFINITION

The surveillance robot is widely used in mostly applications like industrial application (automation), home automation, hospitals, space exploration; military (defence) etc for this purpose continuous surveillance is required. In all the above fields stated there can be hazardous material which can be moved or discarded to safe place.

2.2 EXISTING SYSTEM

In the existing system, robotic arm used in industries is static because it is not integrated with moving vehicle due to which the robot is confined to same place and same work. The surveillance which provides living streaming to detect the obstacles were not present earlier that lead to make use of this robotic vehicle to shortest or limited distance.

2.2.1 LIMITATIONS OF EXISTING SYSTEM

- Robotic arm with the Bluetooth connection covers only shortest distance.
- Without the surveillance there will be no major use of the robot vehicle with the arm because there will lack of knowledge on the obstacles on the path.

To avoid all these limitations and make the working more accurately the system needs to be implemented efficiently

2.3 PROPOSED SYSTEM

Our aim is to control the robotic arm vehicle based on IoT using Arduino. Arduino is used to make the robotic arm vehicle operate through wireless. Robotic arm have DOF along with the gripper. The gripper will pick and place by means of the gear wheels. The base rotates in circular direction and the other two joints for upward, downward and forward, backward motion respectively. There is no limit to the movement each joint can produce since each joint is controlled by a DC motor. The DC motors will be used to control the robotic vehicle. Web camera is plugged into USB port of Arduino where USB ports are available. There are multiple libraries available for camera interface to make it easy to use. The camera module works with all the module of Arduino. Camera is basically used in this system to take real world live videos.

2.3.2 ADVANTAGES OF PROPOSED SYSTEM

- It can be used for military purpose for constant surveillance at border.
- Robotic arm can pick up the items which are hazardous, harmful to humans.
- Robotic arm can reduce the risks of the soldiers by picking up the suspicious items with the help of gripper.
- Robotic arm works in such a way environment where human interaction cannot be possible to do a particular job. .

2.4 FEASIBILITYSTUDY

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require. The following are some of the important financial questions asked during preliminary investigation:

• The costs conduct a full system investigation.

• The cost of the hardware and software.

• The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also, all the resources are already available, it give an indication of the system is economically possible for development.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 BEHAVIORAL FEASIBILITY

This includes the following questions like Is there sufficient support for the users? Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioural aspects are considered carefully and conclude that the project is behaviourally feasible.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS

Hardware interfaces specifies the logical characteristics of each interface between the software product and the hardware components of the system. The following are some hardware requirements.

- Arduino UNO.
- LM298 Motor Driver.
- ESP 32 CAM.
- DC Motors.
- ESP 32 CAM WiFi & Bluetooth Module.
- Servo motors
- Robotic arm and jaw

2.5.2 SOFTWARE REQUIREMENTS

Software Requirements specifies the logical characteristics of each interface and software components of the system. The following are some software requirements,

- Arduino language for coding
- Arduino IDE

3.ARCHITECTURE

3. ARCHITECTURE

3.1 PROJECT ARCHITECTURE

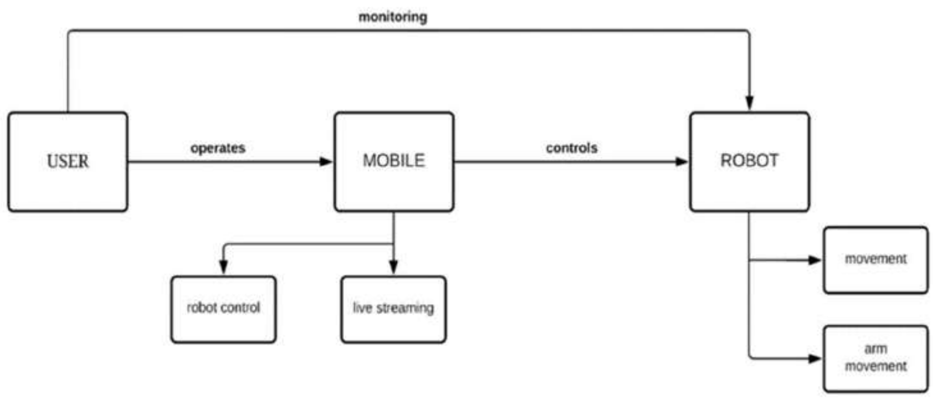


Figure 3.1 Project Architecture for Advanced Surveillance Robot With Robotic Arm Control Over IOT

3.2 DESCRIPTION

The mobile acts as an interface between the user and robot vehicle. The user can operate the robot vehicle through mobile. The user has to connect to Wifi. The robot vehicle consists of an arm ,motor wheels that are connected to DC motors. The L293D motor driver is used to drive these two DC motors in any direction. The ESP 32 CAM is placed on the top of the robot vehicle, which is used for live streaming purposes. This ESP 32 CAM is programmed using Embedded C language. The L293D motor driver is connected to ESP32 Cam to protect it from back EMF. Both ESP32 Cam and L293D motor driver are connected to batteries respectively. As the user uses the control keys, the robot vehicle moves by providing live streaming to the user.

Arduino UNO: Arduino UNO is a credit card-sized chip which has control over every part of the robot. Arduino is a cost-efficient and easy programmable tool. It includes 6 analogue input pins and 14 digital Input/Output pins. The purpose of the Arduino is to provide the robot or board with power.

LM298 Motor Driver: The motor driver L293D is an integrated chip that includes dual-channel H-bridge and pulse-width modulations for rotation and speed using encoded data in pulsing signals. This L298N motor driver can drive two motors with voltages ranging from 4.5 to 36V, allowing it to function as a two-wheel vehicle. It also has capacitors to protect against back EMF produced by motors.

ESP32 CAM: The ESP32 camera module is built around an ESP32-S chip, which can be used as a general-purpose microcontroller. Pre-made modules consist of an actual ESP-32 SoC, external flash memory, and a pre-tuned PCB antenna or an IPEX antenna connector in ESP32-based designs. The ESP32 CAM can be programmed; code is installed via serial pins via an FTDI cable.

ESP32 CAM WiFi & Bluetooth module: ESP 32 CAM WiFi & Bluetooth module is a wireless module which contains a 2Mp camera to recognise objects and wifi or Bluetooth is used to control the robot. The ESP32 is pre-loaded with low-level device drivers and WiFi and Bluetooth wireless protocol stacks. Bluetooth and WiFi can both be used in certain situations. However, we are using WiFi because it has a greater range of coverage.

DC Motors: DC Motors converts electrical energy into Mechanical energy when powered to current. Mostly DC Motors is used for the wheel movement. Other devices that convert electricity to motion do not produce usable mechanical powers, such as electric motors. DC motors produce mechanical power, or torque, which is used to drive the wheels. A servo motor is created by combining a DC motor and a servomechanism.

Servo Motors: Servo Motors convert the electrical energy into Mechanical energy when powered to current. Servo Motors are used to control the angle of rotation. Servo Motors are used to control Arms and Jaw.

Robotic Arm and Jaw: It is a type of mechanical arm. It contains a pair of gripers to mimic the actions of a simple robotic hand. this arm can be controlled manually by associating with a servo motor.

BO Motor: Bo motor is a Battery Operated which is a lightweight DC geared motor that produces the maximum torque and rpm at low voltages.

Motor Driver: Motor drivers bridge the gap between motors and control circuits. The motor requires a large current, but the controller circuit continues to operate on low current signals. The function of motor drivers is to convert a low-current control signal into a higher-current signal capable of moving a motor.

BO Motor Wheel: High quality plastic wheels to mount or to have movement. The wheels have a certain rpm.BO Wheel has a pin to attach the bo motor.

3.3 USE CASE DIAGRAM

Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally.

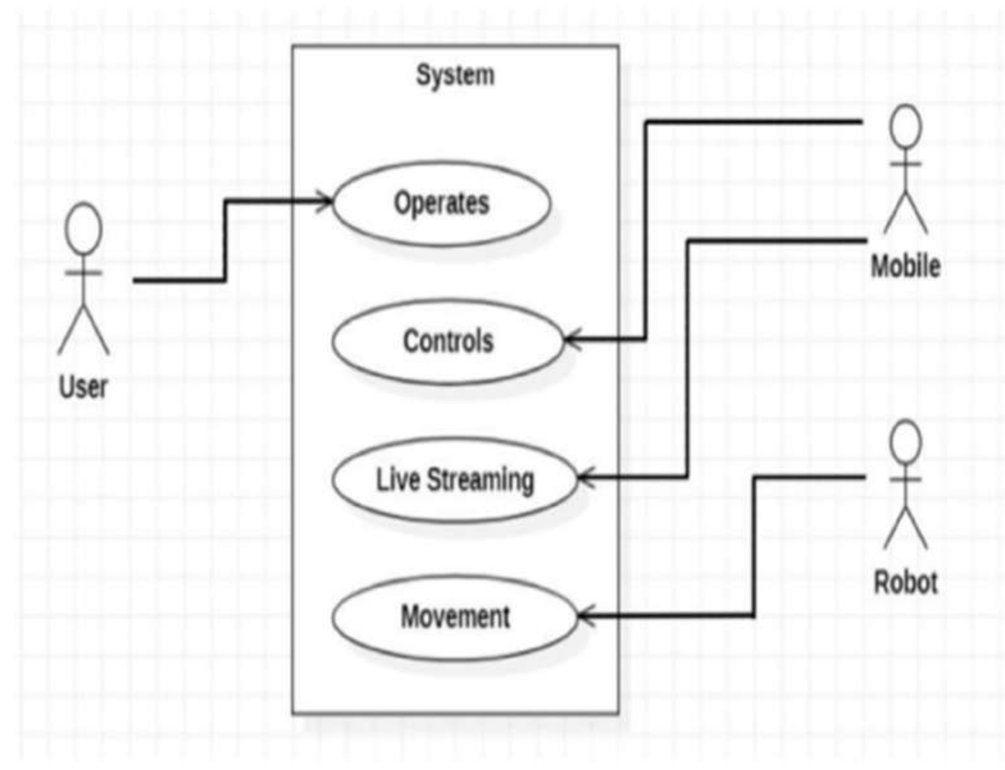


Figure 3.3 Use Case Diagram for Advanced Surveillance Robot With Robotic Arm Control Over IOT

3.4 CLASS DIAGRAM

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity.

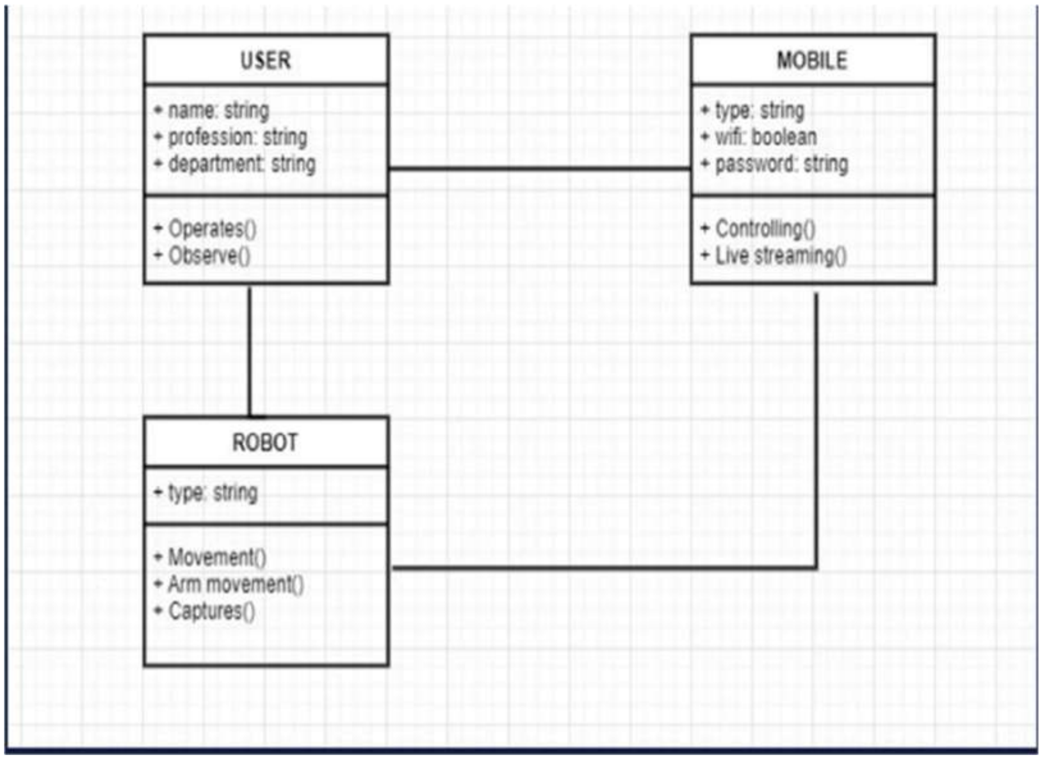


Figure 3.4 Class Diagram for Advanced Surveillance Robot With Robotic Arm Control Over IOT

3.5 SEQUENCE DIAGRAM

A sequence diagram is a type of interaction diagram because it describes how and in what order a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process.

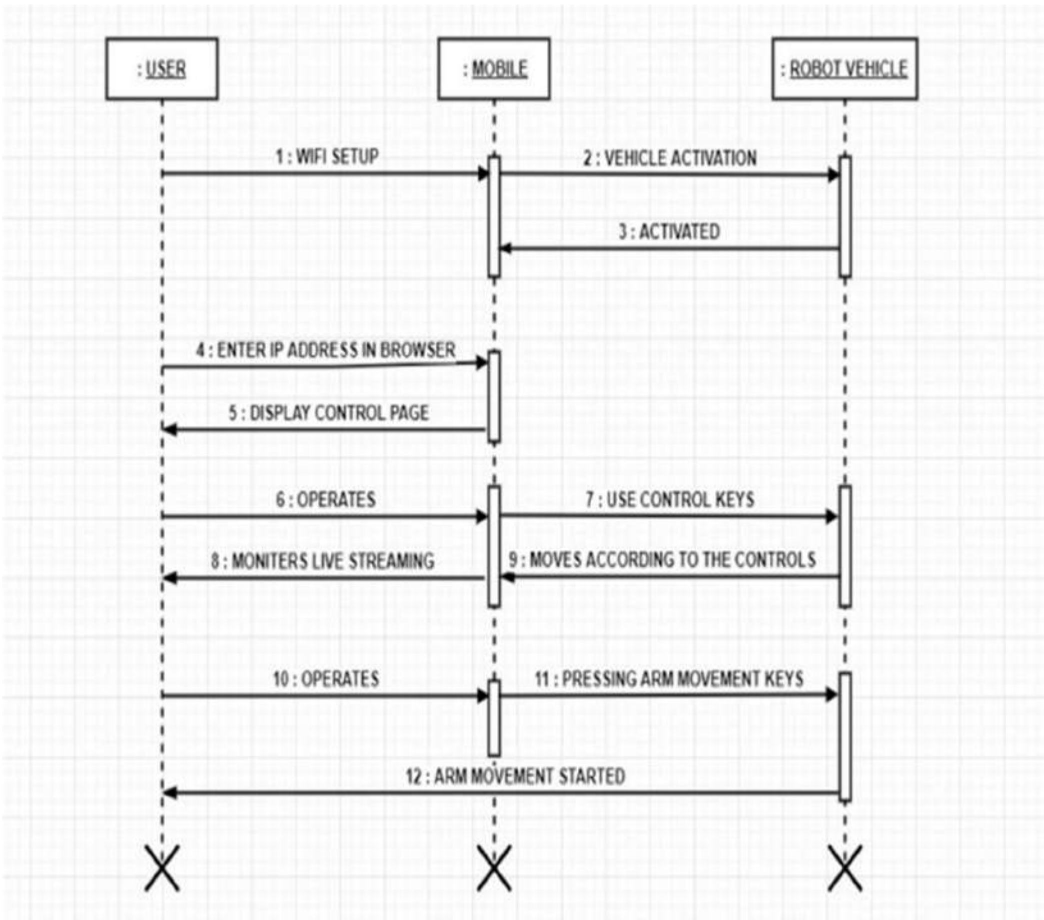


Figure 3.5 Sequence Diagram for Advanced Surveillance Robot With Robotic Arm Control Over IOT

3.6 ACTIVITY DIAGRAM

An activity diagram is **a behavioral diagram i.e. it depicts the behavior of a system**. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

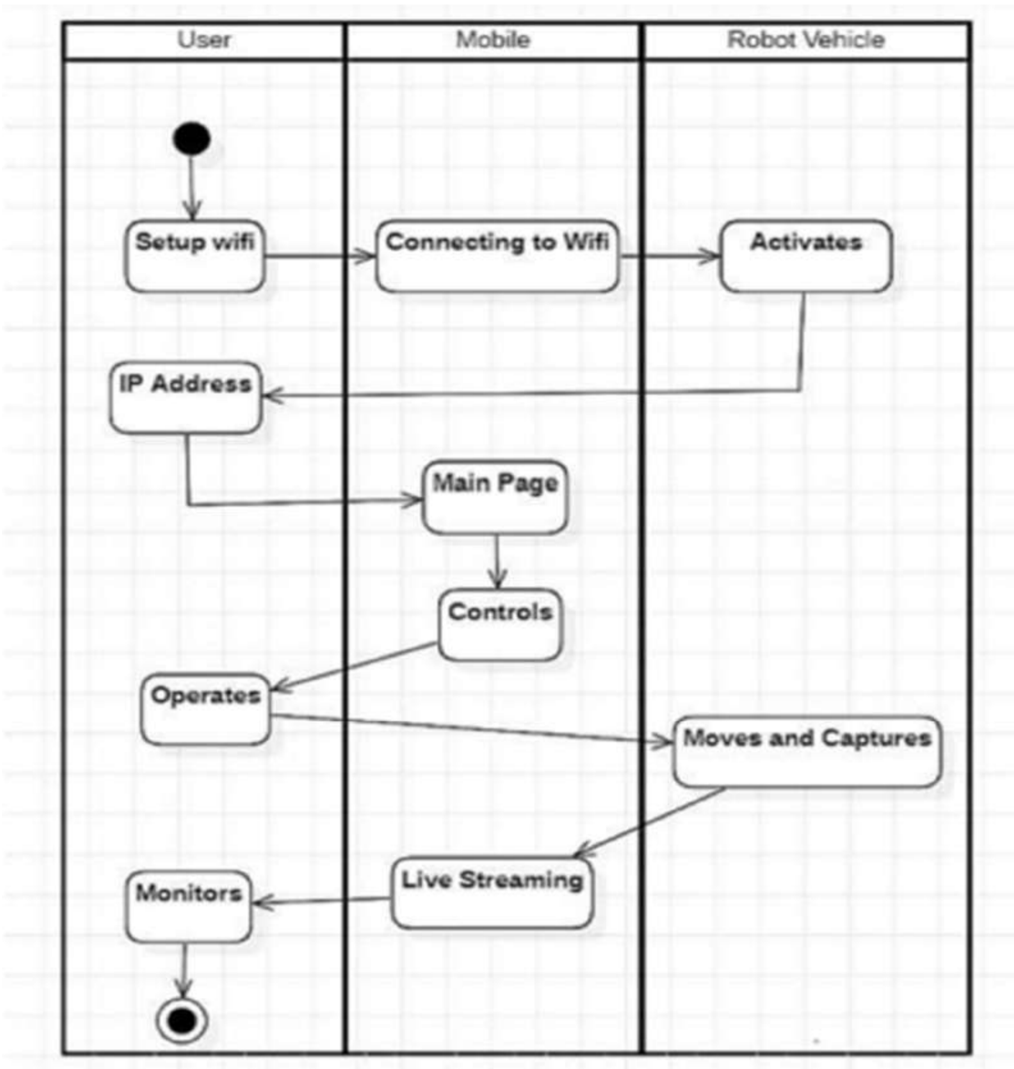


Figure 3.6 Activity Diagram for Recommendation System

4.IMPLEMENTATION

4. IMPLEMENTATION

4.1 SAMPLE CODE

Arduino code

```
#include <Servo.h>


Servo servo1;

Servo servo2;

Servo servo3;

Servo servo4;


int s1pos=90;

int s2pos=90;

int s3pos=40;

int s4pos=120;


String action;


void setup() {


  Serial.begin(115200);

  Serial.println("ready to recieve");

  servo1.attach(3);

  servo2.attach(5);

  servo3.attach(6);
```

```
  servo4.attach(9);


  servo1.write(s1pos);

  servo2.write(s2pos);

  servo3.write(s3pos);

  servo4.write(s4pos);// put your setup code here, to run once:


  delay(2000);

}


void loop() {


  if(Serial.available()>0)

  {

  action=Serial.readStringUntil('\n');


  if(action.indexOf('1')>0)

  {

    Serial.println(s1pos);

    if(s1pos<150)

    { int oldpos=s1pos;

      s1pos+=70;

      for (int pos = oldpos; pos <= s1pos; pos += 1) {

      servo1.write(pos);
```

```
    delay(10);                          }
  }
  delay(100);
}
else if(action.indexOf('2')>0)
{
  Serial.println(s2pos);
  if(s2pos<130)
  { int oldpos=s2pos;
    s2pos+=50;
    for (int pos = oldpos; pos <= s2pos; pos += 1) {
    servo2.write(pos);
    delay(10);                          }
  }
  delay(100);
}
else if(action.indexOf('3')>0)
{
  Serial.println(s3pos);
  if(s3pos<70)
  { int oldpos=s3pos;
    s3pos+=40;
    for (int pos = oldpos; pos <= s3pos; pos += 1) {
    servo3.write(pos);
```

```
    delay(10);                    }
  }
  delay(100);
}
else if(action.indexOf('4')>0)
{
  Serial.println(s4pos);
  if(s4pos<80)
  { s4pos+=55;
    servo4.write(s4pos);
  }
  delay(100);
}
else if(action.indexOf('5')>0)
{
  Serial.println(s1pos);
  if(s1pos>30)
  { int oldpos=s1pos;
    s1pos-=70;
    for (int pos = oldpos; pos >= s1pos; pos -= 1) {
    servo1.write(pos);
    delay(10);                          }
  }
  delay(100);
```

```
}
else if(action.indexOf('6')>0)
{
  Serial.println(s2pos);
  if(s2pos>50)
  { int oldpos=s2pos;
    s2pos-=50;
    for (int pos = oldpos; pos >= s2pos; pos -= 1) {
    servo2.write(pos);
    delay(10);                    }
  }
  delay(100);
}
else if(action.indexOf('7')>0)
{
  Serial.println(s3pos);
  if(s3pos>10)
  { int oldpos=s3pos;
    s3pos-=40;
    for (int pos = oldpos; pos >= s3pos; pos -= 1) {
    servo3.write(pos);
    delay(10);                    }
  }
  delay(100);
```

```
 }
 else if(action.indexOf('8')>0)
 {
   Serial.println(s4pos);
   if(s4pos>110)
   { s4pos-=50;
     servo4.write(s4pos);
   }
   delay(100);
 }
 }
 delay(1000);
}
```

ESP Module code

```
#include "esp_camera.h"
#include <WiFi.h>
#include "esp_timer.h"
#include "img_converters.h"
#include "Arduino.h"
#include "fb_gfx.h"
#include "soc/soc.h"             // disable brownout problems
#include "soc/rtc_cntl_reg.h"    // disable brownout problems
#include "esp_http_server.h"
```

```
// Replace with your network credentials
const char* ssid = "ABCD";
const char* password = "123456789";

#define PART_BOUNDARY "123456789000000000000987654321"

#define CAMERA_MODEL_AI_THINKER
//#define CAMERA_MODEL_M5STACK_PSRAM
//#define CAMERA_MODEL_M5STACK_WITHOUT_PSRAM
//#define CAMERA_MODEL_M5STACK_PSRAM_B
//#define CAMERA_MODEL_WROVER_KIT

#if defined(CAMERA_MODEL_WROVER_KIT)
  #define PWDN_GPIO_NUM    -1
  #define RESET_GPIO_NUM   -1
  #define XCLK_GPIO_NUM    21
  #define SIOD_GPIO_NUM    26
  #define SIOC_GPIO_NUM    27

  #define Y9_GPIO_NUM      35
  #define Y8_GPIO_NUM      34
  #define Y7_GPIO_NUM      39
  #define Y6_GPIO_NUM      36
```

```
  #define Y5_GPIO_NUM      19
  #define Y4_GPIO_NUM      18
  #define Y3_GPIO_NUM       5
  #define Y2_GPIO_NUM       4
  #define VSYNC_GPIO_NUM   25
  #define HREF_GPIO_NUM    23
  #define PCLK_GPIO_NUM    22

#elif defined(CAMERA_MODEL_M5STACK_PSRAM)
  #define PWDN_GPIO_NUM     -1
  #define RESET_GPIO_NUM    15
  #define XCLK_GPIO_NUM     27
  #define SIOD_GPIO_NUM     25
  #define SIOC_GPIO_NUM     23

  #define Y9_GPIO_NUM       19
  #define Y8_GPIO_NUM       36
  #define Y7_GPIO_NUM       18
  #define Y6_GPIO_NUM       39
  #define Y5_GPIO_NUM        5
  #define Y4_GPIO_NUM       34
  #define Y3_GPIO_NUM       35
  #define Y2_GPIO_NUM       32
  #define VSYNC_GPIO_NUM    22
```

```
  #define HREF_GPIO_NUM     26
  #define PCLK_GPIO_NUM     21

#elif defined(CAMERA_MODEL_M5STACK_WITHOUT_PSRAM)
  #define PWDN_GPIO_NUM     -1
  #define RESET_GPIO_NUM    15
  #define XCLK_GPIO_NUM     27
  #define SIOD_GPIO_NUM     25
  #define SIOC_GPIO_NUM     23

  #define Y9_GPIO_NUM       19
  #define Y8_GPIO_NUM       36
  #define Y7_GPIO_NUM       18
  #define Y6_GPIO_NUM       39
  #define Y5_GPIO_NUM        5
  #define Y4_GPIO_NUM       34
  #define Y3_GPIO_NUM       35
  #define Y2_GPIO_NUM       17
  #define VSYNC_GPIO_NUM    22
  #define HREF_GPIO_NUM     26
  #define PCLK_GPIO_NUM     21

#elif defined(CAMERA_MODEL_AI_THINKER)
  #define PWDN_GPIO_NUM     32
```

```
  #define RESET_GPIO_NUM    -1
  #define XCLK_GPIO_NUM      0
  #define SIOD_GPIO_NUM     26
  #define SIOC_GPIO_NUM     27

  #define Y9_GPIO_NUM       35
  #define Y8_GPIO_NUM       34
  #define Y7_GPIO_NUM       39
  #define Y6_GPIO_NUM       36
  #define Y5_GPIO_NUM       21
  #define Y4_GPIO_NUM       19
  #define Y3_GPIO_NUM       18
  #define Y2_GPIO_NUM        5
  #define VSYNC_GPIO_NUM    25
  #define HREF_GPIO_NUM     23
  #define PCLK_GPIO_NUM     22

#elif defined(CAMERA_MODEL_M5STACK_PSRAM_B)
  #define PWDN_GPIO_NUM     -1
  #define RESET_GPIO_NUM    15
  #define XCLK_GPIO_NUM     27
  #define SIOD_GPIO_NUM     22
  #define SIOC_GPIO_NUM     23
```

```
  #define Y9_GPIO_NUM       19
  #define Y8_GPIO_NUM       36
  #define Y7_GPIO_NUM       18
  #define Y6_GPIO_NUM       39
  #define Y5_GPIO_NUM        5
  #define Y4_GPIO_NUM       34
  #define Y3_GPIO_NUM       35
  #define Y2_GPIO_NUM       32
  #define VSYNC_GPIO_NUM    25
  #define HREF_GPIO_NUM     26
  #define PCLK_GPIO_NUM     21

#else
  #error "Camera model not selected"
#endif

#define MOTOR_1_PIN_1    14
#define MOTOR_1_PIN_2    15
#define MOTOR_2_PIN_1    13
#define MOTOR_2_PIN_2    12

static const char* _STREAM_CONTENT_TYPE = "multipart/x-mixed-replace;boundary=" PART_BOUNDARY;
static const char* _STREAM_BOUNDARY = "\r\n--" PART_BOUNDARY "\r\n";
```

```
static const char* _STREAM_PART = "Content-Type: image/jpeg\r\nContent-Length:
%u\r\n\r\n";

httpd_handle_t camera_httpd = NULL;
httpd_handle_t stream_httpd = NULL;

static const char PROGMEM INDEX_HTML[] = R"rawliteral(
<html>
  <head>
    <title>ESP32-CAM Robot</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <style>
      body { font-family: Arial; text-align: center; margin:0px auto; padding-top: 30px;}
      table { margin-left: auto; margin-right: auto; }
      td { padding: 8 px; }
      .button {
        background-color: #2f4468;
        border: none;
        color: white;
        padding: 10px 20px;
        text-align: center;
        text-decoration: none;
        display: inline-block;
        font-size: 18px;
        margin: 6px 3px;
```

```
        cursor: pointer;
        -webkit-touch-callout: none;
        -webkit-user-select: none;
        -khtml-user-select: none;
        -moz-user-select: none;
        -ms-user-select: none;
        user-select: none;
        -webkit-tap-highlight-color: rgba(0,0,0,0);
      }
      img {  width: auto ;
        max-width: 100% ;
        height: auto ;
      }
    </style>
  </head>
  <body>
    <h1>ESP32-CAM Robot</h1>
    <img src="" id="photo" >
    <table>
      <tr><td colspan="3" align="center"><button class="button" onmousedown="toggleCheckbox('forward');" ontouchstart="toggleCheckbox('forward');" onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">Forward</button></td></tr>
      <tr><td align="center"><button class="button" onmousedown="toggleCheckbox('left');" ontouchstart="toggleCheckbox('left');" onmouseup="toggleCheckbox('stop');"
```

```
ontouchend="toggleCheckbox('stop');">Left</button></td><td
align="center"><button class="button" onmousedown="toggleCheckbox('stop');"
ontouchstart="toggleCheckbox('stop');">Stop</button></td><td
align="center"><button class="button" onmousedown="toggleCheckbox('right');"
ontouchstart="toggleCheckbox('right');" onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">Right</button></td></tr>

    <tr><td colspan="3" align="center"><button class="button"
onmousedown="toggleCheckbox('backward');"
ontouchstart="toggleCheckbox('backward');" onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">Backward</button></td></tr>

    <tr><td align="center"><button class="button"
onmousedown="toggleCheckbox('s1f');" ontouchstart="toggleCheckbox('s1f');"
onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">rotateL</button></td><td colspan="3"
align="center"><button class="button" onmousedown="toggleCheckbox('s1b');"
ontouchstart="toggleCheckbox('s1b');" onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">rotateR</button></td></tr>

    <tr><td align="center"><button class="button"
onmousedown="toggleCheckbox('s2f');" ontouchstart="toggleCheckbox('s2f');"
onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">UP</button></td><td colspan="3"
align="center"><button class="button" onmousedown="toggleCheckbox('s2b');"
ontouchstart="toggleCheckbox('s2b');" onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">DOWN</button></td></tr>

    <tr><td align="center"><button class="button"
onmousedown="toggleCheckbox('s3f');" ontouchstart="toggleCheckbox('s3f');"
onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">BACK</button></td><td colspan="3"
align="center"><button class="button" onmousedown="toggleCheckbox('s3b');"
ontouchstart="toggleCheckbox('s3b');" onmouseup="toggleCheckbox('stop');"
ontouchend="toggleCheckbox('stop');">FRONT</button></td></tr>
```

```
<tr><td align="center"><button class="button" onmousedown="toggleCheckbox('s4f');" ontouchstart="toggleCheckbox('s4f');" onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">CLOSE</button></td><td colspan="3" align="center"><button class="button" onmousedown="toggleCheckbox('s4b');" ontouchstart="toggleCheckbox('s4b');" onmouseup="toggleCheckbox('stop');" ontouchend="toggleCheckbox('stop');">OPEN</button></td></tr>
  </table>
 <script>
 function toggleCheckbox(x) {
   var xhr = new XMLHttpRequest();
   xhr.open("GET", "/action?go=" + x, true);
   xhr.send();
 }
 window.onload = document.getElementById("photo").src = window.location.href.slice(0, -1) + ":81/stream";
 </script>
 </body>
</html>
)rawliteral";


static esp_err_t index_handler(httpd_req_t *req){
  httpd_resp_set_type(req, "text/html");
  return httpd_resp_send(req, (const char *)INDEX_HTML, strlen(INDEX_HTML));
}
```

```
static esp_err_t stream_handler(httpd_req_t *req){
  camera_fb_t * fb = NULL;
  esp_err_t res = ESP_OK;
  size_t _jpg_buf_len = 0;
  uint8_t * _jpg_buf = NULL;
  char * part_buf[64];

  res = httpd_resp_set_type(req, _STREAM_CONTENT_TYPE);
  if(res != ESP_OK){
    return res;
  }

  while(true){
    fb = esp_camera_fb_get();
    if (!fb) {
      Serial.println("Camera capture failed");
      res = ESP_FAIL;
    } else {
      if(fb->width > 400){
        if(fb->format != PIXFORMAT_JPEG){
          bool jpeg_converted = frame2jpg(fb, 80, &_jpg_buf, &_jpg_buf_len);
          esp_camera_fb_return(fb);
          fb = NULL;
          if(!jpeg_converted){
```

```
          Serial.println("JPEG compression failed");
          res = ESP_FAIL;
        }
      } else {
        _jpg_buf_len = fb->len;
        _jpg_buf = fb->buf;
      }
    }
  }
  if(res == ESP_OK){
    size_t hlen = snprintf((char *)part_buf, 64, _STREAM_PART, _jpg_buf_len);
    res = httpd_resp_send_chunk(req, (const char *)part_buf, hlen);
  }
  if(res == ESP_OK){
    res = httpd_resp_send_chunk(req, (const char *)_jpg_buf, _jpg_buf_len);
  }
  if(res == ESP_OK){
    res = httpd_resp_send_chunk(req, _STREAM_BOUNDARY,
strlen(_STREAM_BOUNDARY));
  }
  if(fb){
    esp_camera_fb_return(fb);
    fb = NULL;
    _jpg_buf = NULL;
  } else if(_jpg_buf){
```

```
      free(_jpg_buf);
      _jpg_buf = NULL;
    }
    if(res != ESP_OK){
      break;
    }
    //Serial.printf("MJPG: %uB\n",(uint32_t)(_jpg_buf_len));
  }
  return res;
}

static esp_err_t cmd_handler(httpd_req_t *req){
  char*  buf;
  size_t buf_len;
  char variable[32] = {0,};

  buf_len = httpd_req_get_url_query_len(req) + 1;
  if (buf_len > 1) {
    buf = (char*)malloc(buf_len);
    if(!buf){
      httpd_resp_send_500(req);
      return ESP_FAIL;
    }
    if (httpd_req_get_url_query_str(req, buf, buf_len) == ESP_OK) {
```

```
    if (httpd_query_key_value(buf, "go", variable, sizeof(variable)) == ESP_OK) {
    } else {
      free(buf);
      httpd_resp_send_404(req);
      return ESP_FAIL;
    }
  } else {
    free(buf);
    httpd_resp_send_404(req);
    return ESP_FAIL;
  }
  free(buf);
} else {
  httpd_resp_send_404(req);
  return ESP_FAIL;
}

sensor_t * s = esp_camera_sensor_get();
int res = 0;

if(!strcmp(variable, "forward")) {
  //Serial.println("Forward");
  digitalWrite(MOTOR_1_PIN_1, 1);
  digitalWrite(MOTOR_1_PIN_2, 0);
```

```
  digitalWrite(MOTOR_2_PIN_1, 1);
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "right")) {
  //Serial.println("Left");
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 1);
  digitalWrite(MOTOR_2_PIN_1, 1);
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "left")) {
  //Serial.println("Right");
  digitalWrite(MOTOR_1_PIN_1, 1);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 1);
}
else if(!strcmp(variable, "backward")) {
  //Serial.println("Backward");
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 1);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 1);
}
```

```
else if(!strcmp(variable, "stop")) {
  //Serial.println("Stop");
  digitalWrite(MOTOR_1_PIN_1, 0);
  digitalWrite(MOTOR_1_PIN_2, 0);
  digitalWrite(MOTOR_2_PIN_1, 0);
  digitalWrite(MOTOR_2_PIN_2, 0);
}
else if(!strcmp(variable, "s1f")) {
  Serial.println("s1");
}
else if(!strcmp(variable, "s2f")) {
  Serial.println("s2");
}
else if(!strcmp(variable, "s3f")) {
  Serial.println("s3");
}
else if(!strcmp(variable, "s4f")) {
  Serial.println("s4");
}
else if(!strcmp(variable, "s1b")) {
  Serial.println("s5");
}
else if(!strcmp(variable, "s2b")) {
  Serial.println("s6");
```

```
  }
  else if(!strcmp(variable, "s3b")) {
    Serial.println("s7");
  }
  else if(!strcmp(variable, "s4b")) {
    Serial.println("s8");
  }
  else {
    res = -1;
  }

  if(res){
    return httpd_resp_send_500(req);
  }

  httpd_resp_set_hdr(req, "Access-Control-Allow-Origin", "*");
  return httpd_resp_send(req, NULL, 0);
}

void startCameraServer(){
  httpd_config_t config = HTTPD_DEFAULT_CONFIG();
  config.server_port = 80;
  httpd_uri_t index_uri = {
    .uri       = "/",
```

```
  .method    = HTTP_GET,
  .handler   = index_handler,
  .user_ctx  = NULL
};

httpd_uri_t cmd_uri = {
  .uri       = "/action",
  .method    = HTTP_GET,
  .handler   = cmd_handler,
  .user_ctx  = NULL
};
httpd_uri_t stream_uri = {
  .uri       = "/stream",
  .method    = HTTP_GET,
  .handler   = stream_handler,
  .user_ctx  = NULL
};
if (httpd_start(&camera_httpd, &config) == ESP_OK) {
  httpd_register_uri_handler(camera_httpd, &index_uri);
  httpd_register_uri_handler(camera_httpd, &cmd_uri);
}
config.server_port += 1;
config.ctrl_port += 1;
if (httpd_start(&stream_httpd, &config) == ESP_OK) {
```

```
    httpd_register_uri_handler(stream_httpd, &stream_uri);
  }
}

void setup() {
  WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0); //disable brownout detector

  pinMode(MOTOR_1_PIN_1, OUTPUT);
  pinMode(MOTOR_1_PIN_2, OUTPUT);
  pinMode(MOTOR_2_PIN_1, OUTPUT);
  pinMode(MOTOR_2_PIN_2, OUTPUT);

  Serial.begin(115200);
  Serial.setDebugOutput(false);

  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer = LEDC_TIMER_0;
  config.pin_d0 = Y2_GPIO_NUM;
  config.pin_d1 = Y3_GPIO_NUM;
  config.pin_d2 = Y4_GPIO_NUM;
  config.pin_d3 = Y5_GPIO_NUM;
  config.pin_d4 = Y6_GPIO_NUM;
  config.pin_d5 = Y7_GPIO_NUM;
```

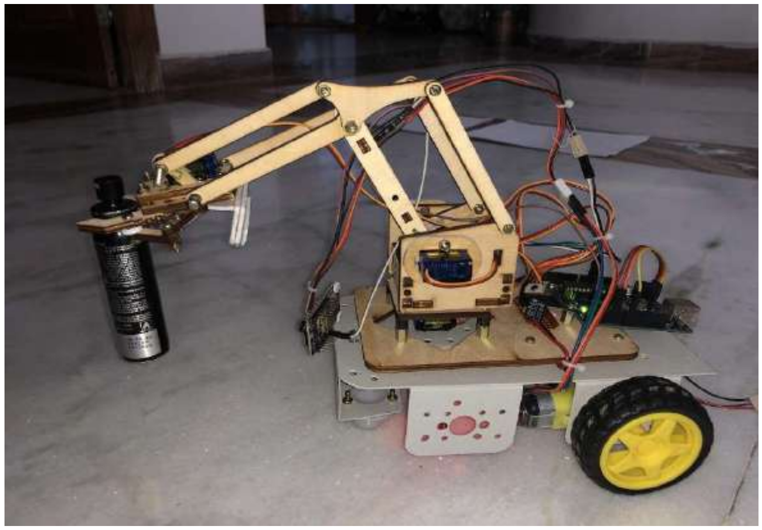
```
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;

if(psramFound()){
  config.frame_size = FRAMESIZE_VGA;
  config.jpeg_quality = 10;
  config.fb_count = 2;
} else {
  config.frame_size = FRAMESIZE_SVGA;
  config.jpeg_quality = 12;
  config.fb_count = 1;
}

// Camera init
```

```
  esp_err_t err = esp_camera_init(&config);
  if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    return;
  }
  // Wi-Fi connection
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");

  Serial.print("Camera Stream Ready! Go to: http://");
  Serial.println(WiFi.localIP());

  // Start streaming web server
  startCameraServer();
}

void loop() {
 }
```

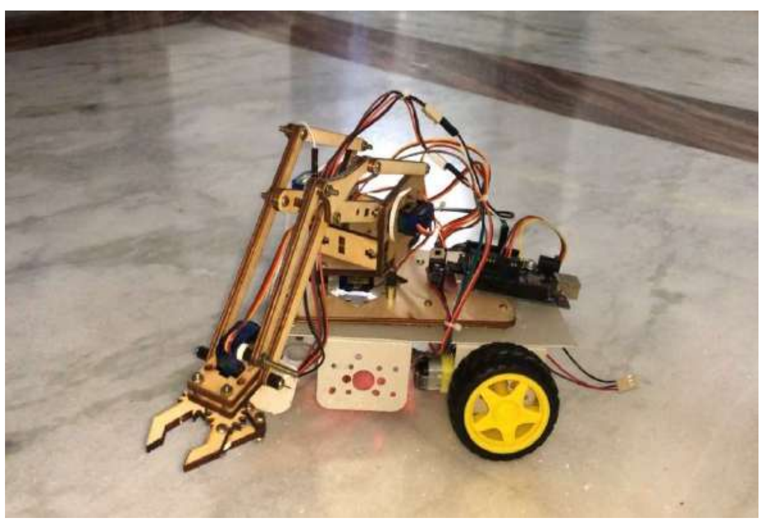
5.SCREENSHOTS

5. SCREENSHOTS

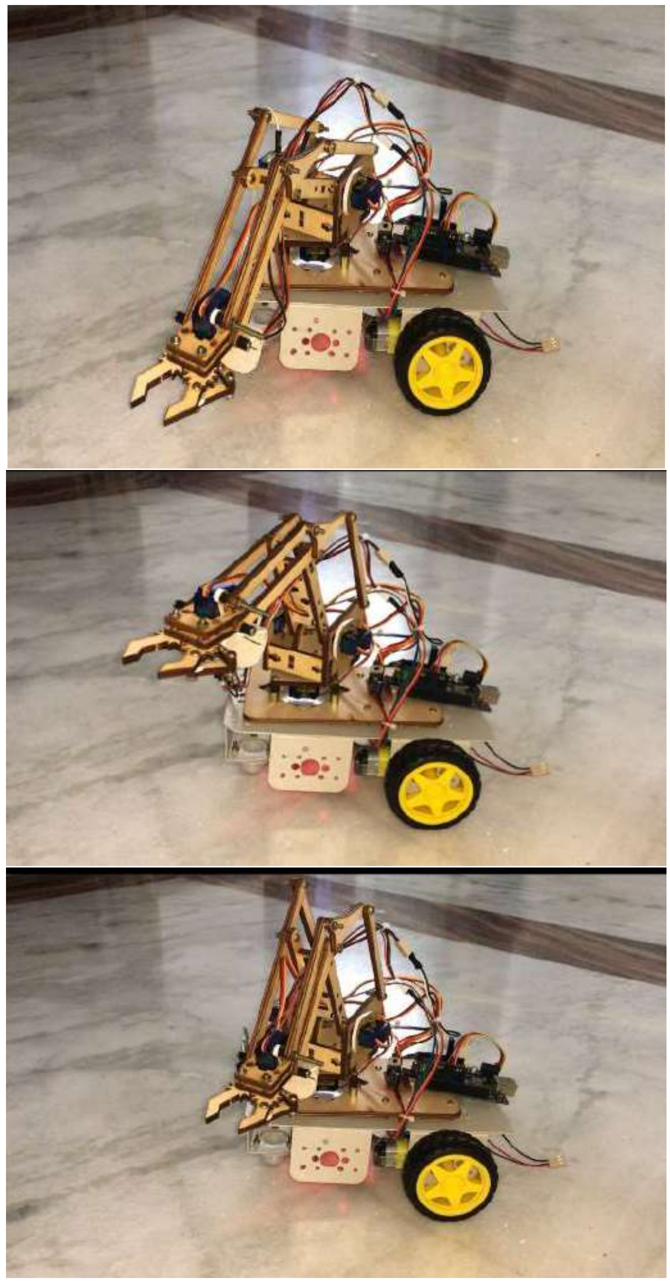
Vehicle



Screenshot 5.1 Vehicle lifting item



Screenshot 5.2 Vehicle



Screenshot 5.3 Designed arm Angle of rotation with Servo Motors

Interface and Controls

Screenshot 5.3 Interface

Screenshot 5.4 vehicle with the control module

6.TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centred on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.
Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows, data fields, predefined processes.

6.3 TEST CASES

Test case ID	Test case name	Purpose	Input	Output
1	Vehicle controls	To check vehicle movement directions.	Pressed forward button.	Vehicle moved forward.
2	Vehicle controls	To check vehicle movement directions.	Pressed right a button.	Vehicle moved towards right.
3	Arm movement	To check robotic arm movement.	Pressed rotateL(rotate left) button.	Arm rotated left.
4	Gripper movement	To check gripper movement	Pressed open button.	Grippers opened.

7.CONCLUSION

7. CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

Although advances in the robotic arms industry have resulted in an increase in idea generation, these devices still fall short of ideal usefulness. Because it can handle multiple jobs at once, this Surveillance gadget can eliminate several jobs that require multiple combinations of equipment. Despite its infancy, our model has already demonstrated its ability to reduce many hazardous situations and develop a more refined approach to traditional methods. More research and development will undoubtedly make the device much more sophisticated and capable of performing a wide range of tasks on its own.

7.2 FUTURE SCOPE

This is the first iteration of our project. In the future, our device’s entire apparatus has the potential to become fully automated and a self-aware system capable of handling much more hazardous debris. The prospects for our device are limitless, as is the range of utility for our device. We hope to expand our design by implementing it in various research facilities and shortly institutes soon.

8.BIBLIOGRAPHY

8. BIBLIOGRAPHY

8.1 REFERENCES

[1] Selvam Muthuraman, “Smart Phone-Based Robotic Control For Surveillance Applications”, International Journal of Research in Engineering and Technology, Volume 03, Issue 03, Mar-2014.

[2] G.Jyothsnadevi, K.Chandisivapriya, B.Saikrishnateja, L.Nagajyothi, U.Dushyanthkumar, “ IOT Controlled Robotic Arm”, International Journal of Scientific & Engineering Research, Volume 10, Issue 03, March-2019.

[3] Kaustubh Gawli, Parinay Karande, Pravin Belose, Tushar Bhadirke, Akansha Bhargava,“ Internet of Things (IoT) Based Robotic Arm”, International Journal of Engineering Research and Technology (IJERT), Volume 5, Issue 01, 2017.

[4] José Eduardo Tzompantzi Netzahual , M.D.S. Higinio Nava Bautista , M.C.C. María Janai Sánchez Hernández,“Real Time Control of Robotic Arm Using Bluetooth Low Energy & Wi-Fi with the Module Board ESP32 and Android Application”, International Journal of Science and Research (IJSR), Volume 08, Issue 04, April-2019.

[5] Kunal Borker & Rohan Gaikwad, “ Wireless Controlled Surveillance Robot”, International Journal of Advanced Research in Computer Science and Management Studies, Volume 2, Issue 2, February-2014 pp. 436-441.

[6] Prathyusha. K, V. Harini, Dr S. Balaji, “DESIGN AND DEVELOPMENT OF A RFID BASED MOBILE ROBOT”, International Journal of Engineering Science & Advanced Technology, Volume 01, Issue 01, December-2011.

8.2 GIT HUB REPOSITORY LINK:

[1] https://github.com/Jayendra-varma/Advanced-surviellance-robot-with-robotic-arm-control.git

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT A DIGITAL LIBRARY

ISSN NO :- 2456-2165

AUTHOR CERTIFICATE

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED

Advance Surveillance Robot with Robotic Arm Control Over IoT

AUTHORED BY

A.Meghana

HAS BEEN PUBLISHED IN

Volume 7 | Issue 6 | June - 2022

ARTICLE DIGITAL NO.

IJISRT22JUN402

EDITOR IN CHIEF IJISRT

A DIGITAL LIBRARY

WWW.IJISRT.COM

IJISRT

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT A DIGITAL LIBRARY

ISSN NO :- 2456-2165

AUTHOR CERTIFICATE

A DIGITAL LIBRARY

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED
Advance Surveillance Robot with Robotic Arm Control Over IoT

AUTHORED BY
D.Jayendra Sai Varma

HAS BEEN PUBLISHED IN
Volume 7 | Issue 6 | June - 2022

ARTICLE DIGITAL NO.
IJISRT22JUN402

EDITOR IN CHIEF IJISRT

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT A DIGITAL LIBRARY

ISSN NO :- 2456-2165

AUTHOR CERTIFICATE

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED

Advance Surveillance Robot with Robotic Arm Control Over IoT

AUTHORED BY

Ashritha Reddy

HAS BEEN PUBLISHED IN

Volume 7 | Issue 6 | June - 2022

ARTICLE DIGITAL NO.

IJISRT22JUN402

EDITOR IN CHIEF IJISRT

A DIGITAL LIBRARY

WWW.IJISRT.COM

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT A DIGITAL LIBRARY

ISSN N0 :- 2456-2165

AUTHOR CERTIFICATE

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED
Advance Surveillance Robot with Robotic Arm Control Over IoT

AUTHORED BY
B.Lalitha

HAS BEEN PUBLISHED IN
Volume 7 | Issue 6 | June - 2022

ARTICLE DIGITAL NO.
IJISRT22JUN402

EDITOR IN CHIEF IJISRT

A DIGITAL LIBRARY

WWW.IJISRT.COM