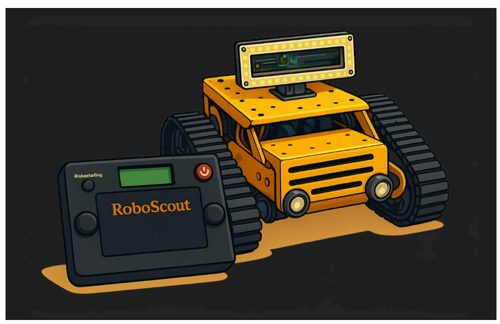
AN-NAJAH NATIONAL UNIVERSITY FACULTY
OF ENGINEERING
AND INFORMATION TECHNOLOGY COMPUTER

ENGINEERING DEPARTMENT




# Hardware Graduation Project

# RoboScout

SUBMITTED BY
**Noor al-huda Hamayel**

**Aseel Maradwi**

SUPERVISOR

**Dr. Mahmoud Assad Dwikat**

Presented in partial fulfillment of the requirements for the Bachelor’s degree in Computer Engineering.

Sept 9, 2025

# Acknowledgment

We would like to express our deepest gratitude to everyone who contributed to the success of this project. First and foremost, we are sincerely thankful to **An-Najah National University**, whose guidance, knowledge, and academic environment have shaped us throughout our years of study and made this achievement possible.

Our heartfelt appreciation goes to our supervisor, **Dr. Mahmoud Assad Dwikat**, for his invaluable guidance, patience, and constant encouragement. His insightful advice, constructive feedback, and unwavering support inspired us to overcome challenges and complete this project with confidence. His mentorship has been a source of motivation and learning that we will carry with us beyond this academic journey.

We are also profoundly grateful to our families, who have been our strongest supporters, offering endless love, prayers, and encouragement during every step of this journey. To our friends and colleagues, thank you for standing by us, sharing knowledge, and lifting our spirits in difficult times.

This work is not only the result of our efforts but also a reflection of the support, inspiration, and dedication we received from all those around us. To each and every one who believed in us — this accomplishment is as much yours as it is ours.

# DISCLAIMER

This report has been written by students of the Computer Engineering Department at An-Najah National University.
It may contain grammatical or technical inaccuracies. The opinions expressed, as well as conclusions and suggestions, are solely those of the students. The university bears no responsibility for consequences resulting from the use of this report beyond its intended academic purpose.

# Contents

# List of Figures

# Abstract

The **RoboScout** project presents the design and implementation of a smart exploration vehicle capable of remote navigation, real-time monitoring, and environmental awareness. The system was developed to address the need for safe and efficient exploration in environments where direct human presence may be difficult or unsafe.

The vehicle is built on a tank chassis for enhanced stability and is powered by an Arduino Mega that controls DC motors, servo motors, and lighting systems. Wireless communication between the handheld controller and the vehicle is achieved using ESP32 modules with ESP-NOW protocol, enabling low-latency transmission of joystick commands and sensor feedback. A Raspberry Pi 4, equipped with Raspberry Pi Camera v2 and a microphone, provides live video and audio streaming to the operator's mobile device.

Multiple sensors were integrated to enhance safety and monitoring, including an ultrasonic sensor for obstacle detection, a DHT11 for temperature and humidity measurement, an MQ sensor for gas detection, an MPU6050 gyroscope for tilt monitoring, and an INA219 sensor for battery level tracking. The system also features a timed safety mechanism: when an obstacle is detected within 20 cm, forward motion is blocked for 10 seconds while reverse movement remains allowed.

To improve usability, an LCD with I2C interface and a buzzer on the handheld controller provide immediate feedback, while headlights and an LED strip automatically adjust to ambient light conditions for improved visibility. A custom 3D-printed enclosure houses the controller, which includes dual joysticks for navigation and camera control. Additionally, a mobile application allows alternative vehicle control and real-time monitoring of sensor data.

The results demonstrate that the **RoboScout** is a reliable, cost-effective, and user-friendly platform that successfully integrates robotics, sensing, and multimedia streaming into a compact system suitable for exploration, monitoring, and educational applications.

# Introduction

Exploration vehicles are essential in applications such as search and rescue, hazardous environment inspection, and educational robotics. However, commercially available systems are often expensive, limited in flexibility, and lack integration of environmental sensing with real-time video and mobile control.

This project proposes a **smart exploration vehicle (ExploRover)** that addresses these limitations by combining affordable hardware components with effective embedded system design. The vehicle is designed to navigate rough terrains, avoid obstacles, provide real-time environmental monitoring, and stream live video and audio to the operator via both hardware controller and mobile application.

## Statement of the problem

Traditional remote-controlled cars provide only basic movement capabilities without environmental awareness or safety measures. In exploration tasks, the absence of obstacle detection, intelligent safety responses, and live environmental data reduces reliability.

Therefore, there is a need for a **cost-effective exploration vehicle** that:

- Provides real-time video and audio feedback.
- Monitors environmental conditions such as temperature, gas, tilt, and battery.
- Enhances safety using ultrasonic obstacle detection with timed stop logic.
- Uses intelligent lighting systems to ensure visibility in low-light conditions.
- Offers both hardware joystick control and a mobile application for flexibility.

## Objectives

This project aims to develop a smart exploration vehicle (**RoboScout**) that addresses the needs of remote monitoring, safe navigation, and environmental awareness. It aims to achieve the following objectives:

1. **Reliable remote control of movement and vision**: The system enables wireless control of the vehicle and its mounted camera using dual joysticks in the handheld controller, with additional redundancy provided through a mobile phone application that allows remote driving and monitoring of sensor data.

2. **Enhanced obstacle detection and safety measures**: The vehicle integrates ultrasonic sensors with buzzer alerts to notify the operator when approaching obstacles. Additionally, a timed safety feature halts forward movement for 10 seconds when the vehicle encounters an obstacle, while still allowing reverse navigation to ensure maneuverability and prevent collisions.

3. **Comprehensive environmental monitoring**: The system incorporates multiple sensors—including DHT11 for temperature and humidity, MQ gas sensor for hazardous gas detection, MPU6050 gyroscope for tilt monitoring, and INA219 for battery level measurement. Together, these provide the operator with real-time awareness of environmental conditions and system health. These values are accessible through both the handheld controller LCD and the mobile application.

4. **High-quality video and audio streaming**: A Raspberry Pi 4 equipped with Raspberry Pi Camera v2 and an external microphone provides live video and audio feeds to the operator's mobile device. This ensures clear situational awareness and supports safe navigation in real time.

5. **Smart automatic lighting**: The vehicle is equipped with headlights and an LED strip that are controlled automatically using an LDR sensor. This ensures improved visibility in dark environments without unnecessary energy consumption in well-lit areas.

6. **User feedback and awareness**: Real-time data—including sensor values, obstacle warnings, and battery percentage—are displayed on both the LCD with I2C interface (in the handheld controller) and within the mobile application. This dual feedback mechanism enhances operator awareness and improves overall system reliability.

7. **Cost-effective and practical solution**: By utilizing affordable and accessible hardware components such as Arduino Mega, ESP32 modules, Raspberry Pi, and common sensors, the system delivers advanced functionality at a fraction of the cost of commercial exploration robots, making it suitable for educational, experimental, and practical use cases.

By achieving these objectives, the project delivers a reliable, user-friendly, and affordable exploration vehicle capable of performing remote monitoring, safe navigation, and environmental sensing in a wide range of conditions.

## Scope of the Work

This project covers the following scope:

- Design and implementation of a vehicle platform controlled by dual joysticks or a mobile application.
- Integration of sensors for obstacle detection, environmental monitoring, tilt, and battery status.
- Implementation of ESP-NOW wireless communication between the controller and the vehicle.
- Real-time video and audio streaming from Raspberry Pi to a mobile device.
- Automatic lighting system controlled by an LDR sensor.
- Development of software for Arduino Mega, ESP32, and a mobile app to handle communication, motor control, sensor fusion, and user feedback.

The project does not cover:

- Autonomous navigation using AI or advanced path planning.
- Long-range communication beyond ESP-NOW's capabilities.
- Industrial-grade vehicle mechanics.

## Significance or Importance of the Work

This project is significant because it:

- Provides a **mid-cost exploration platform** with advanced monitoring and control features.
- Enhances **safety** by integrating ultrasonic detection with timed stop logic.
- Improves **usability** by supporting both hardware joysticks and mobile app control.
- Promotes **learning** in embedded systems, IoT, and robotics.
- Can be adapted for **search and rescue**, **hazard detection**, and **educational use**.

# Constraints and earlier work

## Constraints

During the design and development of the exploration vehicle, several constraints were encountered:

- **Cost**: Focus on using affordable components instead of advanced industrial-grade parts.
- **Time:** The project was developed in just two months.
- **Power Limitations**: Servo motors required stable power to avoid jitter.
- **Communication Range**: ESP-NOW is reliable in medium-range environments but unsuitable for long-range control.
- **ESP Reliability**: One of the most challenging aspects was working with the ESP32 modules, as they occasionally stopped responding or even burned out due to sensitivity in power handling and improper wiring during early tests. This required extra replacements and careful handling, which added complexity and time to the project.

## Earlier work

The successful development and implementation of the **RoboScout exploration vehicle** project were strongly supported by the knowledge and skills gained through various key courses completed by the project team. The most relevant coursework includes:

1. **Electronics and Circuit Design**:
   This course provided essential knowledge about electronic components, circuit design principles, and both analog and digital systems. The understanding gained was crucial for designing the vehicle's hardware circuitry, including power distribution, motor driver interfacing, sensor connections, and the LCD and buzzer alert systems that provide real-time feedback to the operator.
2. **Microprocessors**:
   This course equipped the team with the ability to program microprocessors and interface them with external modules and sensors. It covered fundamental topics such as I/O operations, data communication, and peripheral control. This knowledge was directly applied in programming the Raspberry Pi for live video/audio streaming and the Arduino Mega for controlling motors, sensors, and actuators.
3. **Microcontrollers**:
   The Microcontrollers course focused on advanced concepts in system modeling, feedback control, and real-time embedded systems. Skills gained from this course were vital in designing and implementing the control algorithms for DC motors, servo motors, and sensor fusion logic that ensured smooth operation and reliable vehicle performance.

By leveraging the theoretical and practical knowledge acquired from these earlier courses, the team was able to effectively address real-world engineering challenges, integrate hardware and software components, and deliver a reliable and functional exploration vehicle capable of remote navigation, live monitoring, and environmental sensing.

## Literature review

In this chapter, we review relevant literature and previous works that contributed to the understanding and development of smart exploration vehicles. This review demonstrates our familiarity with the topic and provides a background foundation for the design and implementation of our system.

Autonomous and semi-autonomous vehicles have been an active research field for decades, with particular focus on obstacle detection, environmental sensing, and safe navigation. Ultrasonic sensors have been widely used in robotics due to their simplicity and effectiveness in detecting nearby obstacles. Studies show that ultrasonic-based systems are reliable in short-range detection but must be complemented with safety logic to prevent collisions. Inspired by these findings, our project integrates ultrasonic sensors with a timed stop mechanism that halts forward motion for 10 seconds when obstacles are encountered, while still allowing reverse navigation to ensure safety.

Environmental monitoring in robotics has also gained importance, especially in hazardous exploration. Sensors such as **DHT11 for temperature and humidity**, **MQ sensors for gas detection**, and **INA219 for battery monitoring** have been used in IoT-based systems to provide real-time awareness of surroundings. These sensors enhance decision-making and improve user safety by providing continuous feedback. Our project adopts these sensors and displays their values on both an LCD and a mobile application, ensuring that the operator remains informed at all times.

Modern robotics literature also emphasizes the importance of reliable communication systems. **ESP-NOW**, developed by Espressif, has been validated in several IoT and robotics projects as a low-latency, energy-efficient, and peer-to-peer communication protocol. Unlike Wi-Fi or Bluetooth, ESP-NOW eliminates the need for an access point, making it ideal for direct, short-range communication between the controller and the vehicle. This inspired our choice to implement ESP-NOW for transmitting joystick data and receiving sensor feedback.

For real-time video and audio, previous works have demonstrated the effectiveness of using **Raspberry Pi with camera modules** to provide live streaming in robotics applications. Tutorials and research highlight the flexibility of Raspberry Pi as it can run lightweight servers for transmitting video streams to mobile devices. We adopted this approach to enable the operator to visualize the environment remotely while also hearing sounds from the vehicle's surroundings through a microphone.

Microcontrollers such as the **Arduino Mega** are widely used in robotics projects because of their large number of input/output pins and compatibility with different sensors and actuators. Online tutorials, such as the "Arduino DC Motor Control Tutorial — L298N | PWM | H-Bridge," provide guidance on driving DC motors, implementing PWM for speed control, and managing power to avoid overheating. These insights were crucial in designing the motor control logic of our vehicle.

By synthesizing findings from these studies, tutorials, and prior implementations—including communication protocols, motor control techniques, video streaming, and environmental sensing—we developed a reliable, safe, and user-friendly exploration vehicle. The **RoboScout** combines the lessons of earlier works into a compact, affordable, and practical system tailored for exploration, monitoring, and educational purposes.

## References


1. Raspberry Pi Foundation. *Getting Started with Raspberry Pi.* Available online: https://www.raspberrypi.com/documentation/computers/getting-started.html

2. Raspberry Pi Foundation. *Raspberry Pi OS Documentation.* Available online: https://www.raspberrypi.com/documentation/computers/os.html

3. Arduino Forum. *Communication between Arduino Mega and Raspberry Pi using GPIO TX/RX pins.* Available online: https://forum.arduino.cc/t/communication-between-arduino-mega-and-raspberry-pi-using-gpio-tx-rx-pins/1289900

4. W. Parker. *How to Connect Raspberry Pi and Arduino — Step by Step Tutorial.* YouTube, 2021. Available online: https://www.youtube.com/watch?v=xc9rUI0F6Iw

5. Tom's Hardware. *How to Use Raspberry Pi with Arduino: Step-by-Step Guide.* Available online: https://www.tomshardware.com/how-to/use-raspberry-pi-with-arduino

# Methodology

## Overview of the System

The system we developed is an **exploration vehicle** that enables the operator to remotely control movement, camera direction, and lighting, while also receiving real-time feedback from environmental sensors and live video/audio streaming. The system is composed of three main parts that work together:

1. **Vehicle Unit**: Built with Arduino Mega, ESP32, multiple sensors, motor drivers, servo motors, and a Raspberry Pi camera module.

2. **Controller Unit**: Includes an ESP32 with dual joysticks, an LCD screen, a buzzer, and a tilt-warning LED for real-time operator feedback.

3. **Mobile Application**: Provides alternative control of the vehicle and displays real-time sensor data for added reliability.

## Subsystems

**• Motion Control**
The vehicle's motion is controlled using either the joystick unit or the mobile application. Joystick signals are sent from the controller's ESP32 via ESP-NOW to the vehicle's ESP32, which forwards them to the Arduino Mega. The Arduino then drives the DC motors using an L298N motor driver. Safety logic is implemented so that if the vehicle encounters an obstacle within 20 cm, forward motion is blocked for 10 seconds, but reverse movement remains allowed. After this timeout, the operator can resume forward movement to bypass the obstacle.

**• Camera Control**
The exploration vehicle is equipped with a camera mounted on two servo motors that allow vertical and horizontal movement. The operator can control the camera orientation using the second joystick on the handheld controller.

**• Obstacle Detection & Safety Alerts**
An ultrasonic sensor mounted on the vehicle continuously measures the distance to obstacles. When an obstacle is detected within 40 cm, a buzzer located in the controller unit alerts the operator with variable beeping intervals depending on distance. In addition, if the tilt angle of the vehicle exceeds a defined threshold, detected by the MPU6050 gyroscope, a warning LED on the controller begins flashing to signal unstable terrain.

**• Environmental Monitoring**
The vehicle integrates multiple environmental sensors:

- **DHT11** for measuring temperature and humidity.

- **MQ gas sensor** for detecting harmful gases.

- **INA219** sensor for monitoring battery voltage and displaying charge percentage.

These sensor values are transmitted from the vehicle ESP32 to the controller ESP32 and displayed on the LCD screen. They are also viewable on the mobile application, ensuring the operator always has awareness of environmental conditions.

**• Lighting System**
To ensure visibility in dark environments, the vehicle includes headlights and an LED strip mounted on top. An LDR (light-dependent resistor) detects the surrounding light intensity. If the environment is dark, the Arduino Mega automatically turns on the headlights and LED strip; if sufficient light is present, they remain off to save power.

**• Video and Audio Streaming**
A Raspberry Pi with a connected camera module and microphone handles real-time video and audio streaming. The Pi is configured as an RTSP server, allowing the operator's mobile device to view the live feed directly through a streaming application. This setup enables both visual navigation and audio awareness of the vehicle's surroundings.

**• Mobile Application**
In addition to the physical joystick controller, a custom mobile application was developed to provide alternative control. Through the app, the operator can drive the vehicle and view live sensor readings in real time. This redundancy ensures reliable operation even if the joystick controller is unavailable.

## Hardware components and implementation

Here is a list of the hardware components used to build the exploration vehicle and controller system:

- **Arduino Mega**

  We used the Arduino Mega as the main controller for the vehicle. It was connected to the DC motors through the L298N driver, to the servo motors for the camera, and to the lighting system (headlights and LED strip). It also managed the automatic lighting logic by reading the LDR sensor. The Mega was chosen due to its large number of pins and stable handling of multiple peripherals simultaneously.

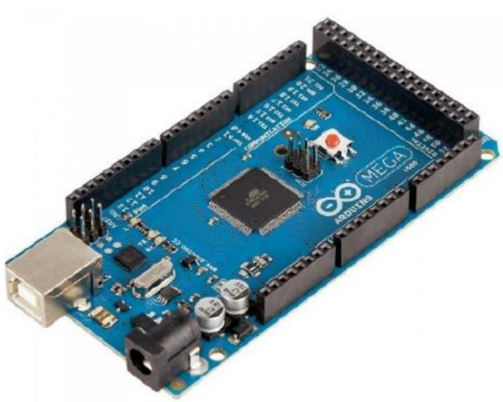


Figure 1:Arduino Mega

- **ESP32 Modules**

  Two ESP32 modules were used: one in the handheld controller and the other in the vehicle. They communicate via ESP-NOW to exchange joystick commands and sensor readings in real time. Additionally, the ESP32 is responsible for forwarding sensor data to the LCD and buzzer on the controller side.

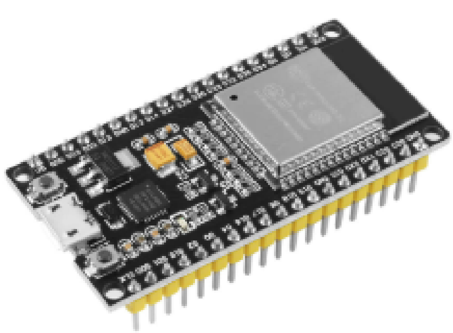


Figure 2:ESP32 Modules

- **L298N Motor Driver**

  We used the L298N H-Bridge driver to control the two DC motors that drive the vehicle's wheels. It allowed both direction control and PWM speed adjustment. Safety logic was added to prevent forward movement when obstacles are detected.

Figure 3:L298N Motor Driver

- **Servo Motors**
  Two servo motors were mounted to hold the camera: one for horizontal rotation and one for vertical tilt. This configuration allows the operator to freely adjust the camera's direction using the joystick or mobile app.


Figure 4:Servo Motors

- **Ultrasonic Sensor (HC-SR04)**
  The ultrasonic sensor is used for obstacle detection. If the vehicle comes within 20 cm of an object, the Arduino automatically stops forward movement for 10 seconds. During this time, the operator can still move the vehicle backward or sideways.

Figure 5:Ultrasonic Sensor (HC-SR04)

- **DHT11 Sensor**
  The DHT11 sensor measures both temperature and humidity in the surrounding environment. These values are transmitted to the handheld controller and displayed on the LCD, ensuring the operator is aware of environmental conditions.

Figure 6:DHT11 Sensor

- **MQ Gas Sensor**
  The MQ sensor is used to detect the presence of harmful gases. Its readings are averaged and filtered before being displayed on the controller's LCD.


Figure 7:MQ Gas Sensor

- **MPU6050 Gyroscope**
  The gyroscope continuously monitors the tilt of the vehicle. If the tilt angle exceeds a safe threshold, a red LED on the handheld controller begins to flash, alerting the operator of unstable terrain.

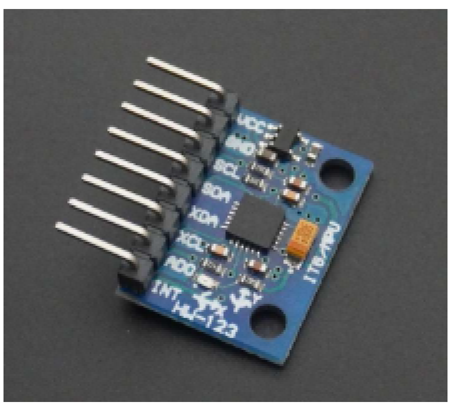


Figure 8:MPU6050 Gyroscope

- **INA219 Power Sensor**
  The INA219 module monitors the battery's voltage and current, providing a real-time percentage of charge. This is crucial for field operation to avoid sudden power loss. The readings are sent to the controller's LCD.

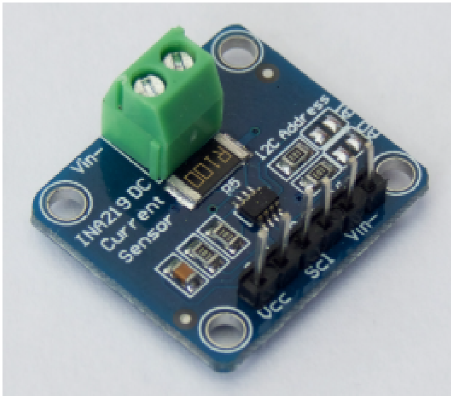



Figure 9:INA219 Power Sensor

- **LCD 16x2 with I2C Interface**
 We used a 16x2 LCD with an I2C interface, connected to the controller ESP32. The I2C interface significantly reduced the number of required pins, making wiring simpler and more reliable compared to parallel connections. The LCD displays live values of temperature, humidity, gas levels, distance to obstacles, and battery percentage. This ensures the operator has continuous situational awareness without overloading the hardware with complex wiring.

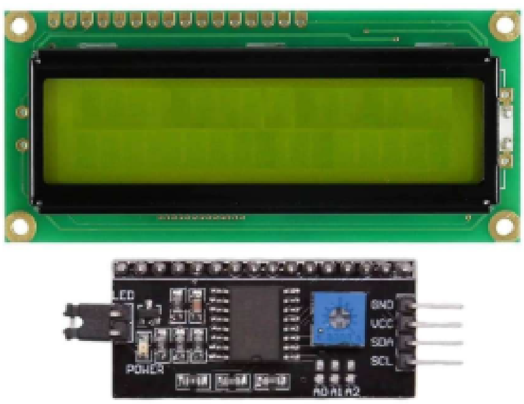



Figure 10:LCD 16x2 with I2C Interface

- **Buzzer and LED Indicators**
A buzzer on the controller provides audio alerts when the vehicle approaches an obstacle. The tone interval changes according to distance. LEDs also indicate tilt warnings and power status.

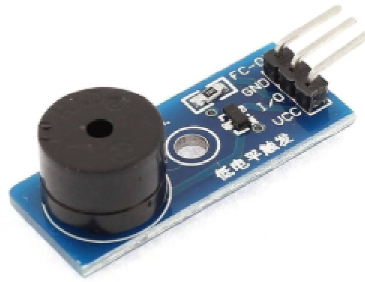


Figure 12:buzzer

Figure 11:LED

- **LED Strip and Headlights**
  For navigation in dark environments, the vehicle is equipped with headlights and an LED strip mounted on top. These lights are controlled by an LDR sensor: they turn on automatically in dark conditions and remain off when ambient light is sufficient.

Figure 13:LED Strip

Figure 14:Headlights

- **Raspberry Pi 4 with Raspberry Pi Camera v2 and Microphone**
  The Raspberry Pi 4 was used as the multimedia and streaming unit of the exploration vehicle. It was equipped with the Raspberry Pi Camera v2 for high-quality video capture and an external microphone for real-time audio recording. Together, these components enabled live video and audio streaming through an RTSP server hosted on the Raspberry Pi. The operator could access the stream directly from a mobile device, allowing clear visual navigation of the environment as well as awareness of surrounding sounds. This integration provided a reliable and efficient solution for remote monitoring in exploration scenarios.

Figure 15:Raspberry Pi 4

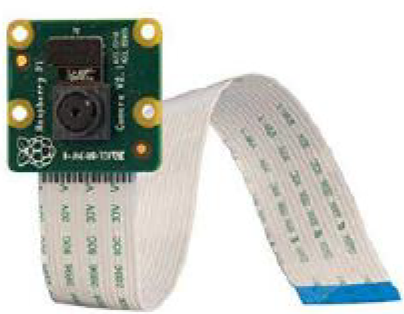


Figure 16:Rasbperry cam v2

Figure 17:Microphone

- **Dual Joysticks**
  Two analog joysticks were used in the handheld controller. The first joystick controls the vehicle's movement (forward, backward, left, right), while the second joystick controls the camera's orientation (pan and tilt). These joysticks provide the operator with intuitive and precise control of both navigation and vision.

Figure 18:Joystick

- **Tank Chassis**
  The vehicle was built on a tank chassis, which provided stability and better traction on rough terrain compared to a regular wheeled frame. The tank base allowed the exploration vehicle to move in different environments while maintaining balance and durability.

Figure 19:Tank Chassis

- **3D-Printed Controller Enclosure**
  The handheld controller (joysticks, LCD, buzzer, LEDs, and ESP32) was housed in a custom 3D-printed case. This ensured proper ergonomics for the operator, protected the electronics, and gave the controller a professional and portable design.


Figure 20:3D-Printed Controller

**DC Motors**

Two high-torque DC motors were used to drive the vehicle's tracks through the L298N motor driver. The motors were carefully selected to provide enough power for movement across different surfaces while still being compatible with the power supply system of the vehicle.

Figure 21:DC Motors

## Software Application

We developed a mobile application to provide an alternative control interface in addition to the physical joystick controller. The app allows the operator to:

- Drive the vehicle (forward, backward, left, right).
- Monitor live sensor readings, including temperature, humidity, gas levels, distance to obstacles, tilt warnings, and battery percentage.
- Access real-time data in parallel with the LCD on the handheld controller, ensuring redundancy and reliability.
- View live video and audio streams from the Raspberry Pi through a compatible streaming application.

This mobile application ensures flexibility and reliability: if the joystick controller is unavailable or malfunctions, the vehicle can still be controlled and fully monitored through the app.

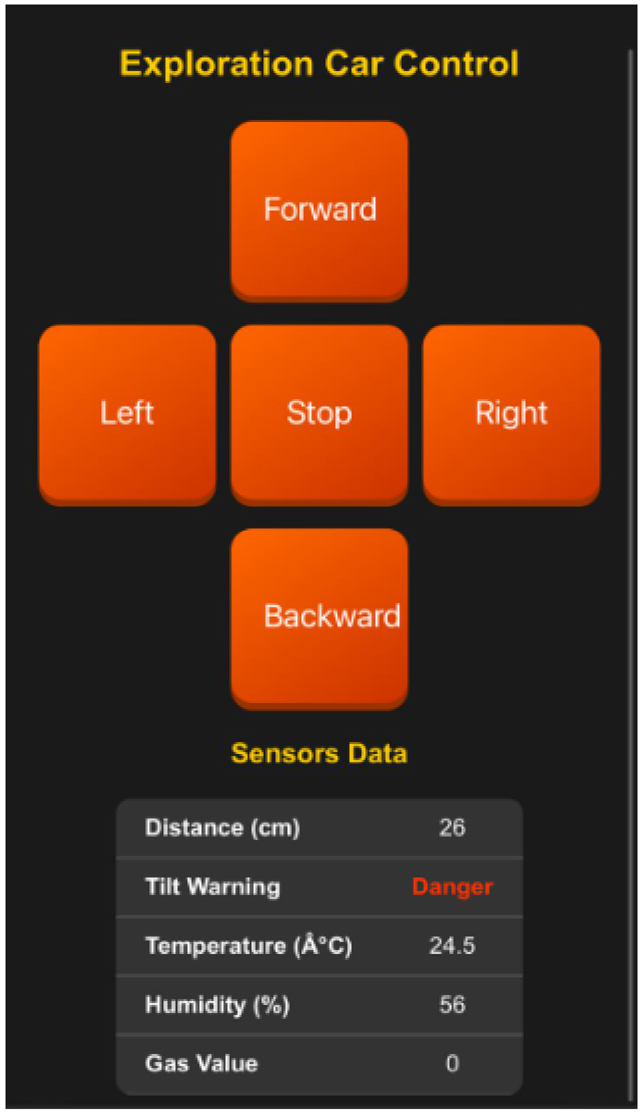



Figure 22:App

## Results and conclusions

The developed exploration vehicle (**RoboScout** ) successfully achieved its core objectives, demonstrating both technical feasibility and practical usefulness in remote navigation and monitoring.

Through the integration of an Arduino Mega–based control system, ESP32 wireless communication, and a Raspberry Pi multimedia module, the vehicle was able to:

- **Provide reliable navigation control** using dual joysticks in the handheld controller, with full redundancy through a mobile application for remote driving and monitoring of sensor data.

- **Enhance safety** by integrating ultrasonic sensors and obstacle-avoidance logic that halts forward movement for 10 seconds when an obstacle is detected, while still allowing reverse navigation to ensure operator control.

- **Deliver real-time environmental monitoring**, including temperature, humidity, gas concentration, tilt warnings, and battery status, with values displayed on both the LCD screen (via I2C interface) and the mobile application.

- **Enable live video and audio streaming** through the Raspberry Pi 4 with Camera v2 and microphone, providing the operator with clear situational awareness.

- **Implement smart lighting control**, where headlights and an LED strip are automatically activated in low-light conditions based on LDR readings, improving visibility without wasting energy.

- **Offer ergonomic and professional design**, using a tank chassis for stability and a custom 3D-printed enclosure for the handheld controller, ensuring durability and ease of use.

This project delivers a practical and innovative solution for exploration and monitoring tasks. By combining robotics, environmental sensing, wireless communication, and multimedia streaming, the **RoboScout** achieves a strong balance between cost, functionality, and reliability—making it a valuable platform for both educational and applied exploration purposes.


## Future work

While the **RoboScout** exploration vehicle achieved its intended objectives, there are several potential improvements and extensions that could enhance its performance, reliability, and application scope in future work:

- **Integration of GPS and Mapping**: Adding a GPS module and mapping software would allow the vehicle to record its path, provide location tracking, and even enable semi-autonomous navigation.

- **Long-Range Communication**: Currently, ESP-NOW provides reliable short-to-medium range control. Future upgrades could include using LoRa or 4G/5G modules to extend communication range for outdoor exploration.

- **Enhanced Power System**: Incorporating larger batteries or a solar charging system would extend operation time and make the vehicle more suitable for long missions.

- **AI-Based Object Detection**: Using machine learning algorithms on the Raspberry Pi, the system could recognize specific objects or hazards (such as fire, smoke, or humans), making it more autonomous in rescue or hazardous environments.

- **Data Logging and Cloud Integration**: Storing environmental sensor data on a cloud server would allow real-time remote monitoring and historical analysis, enabling broader applications in research and safety.

By pursuing these improvements, the **RoboScout** could evolve from a semi-manual exploration tool into a more advanced, semi-autonomous or fully autonomous platform, suitable for diverse real-world applications such as environmental monitoring, disaster response, and research in hazardous areas.