

A detailed photograph of a breadboard circuit. An ESP32 microcontroller is the central component, with numerous jumper wires connecting it to other parts of the circuit. A red LED is visible on the right side of the breadboard. The breadboard is placed on a black keyboard. The background is a light-colored, textured surface.

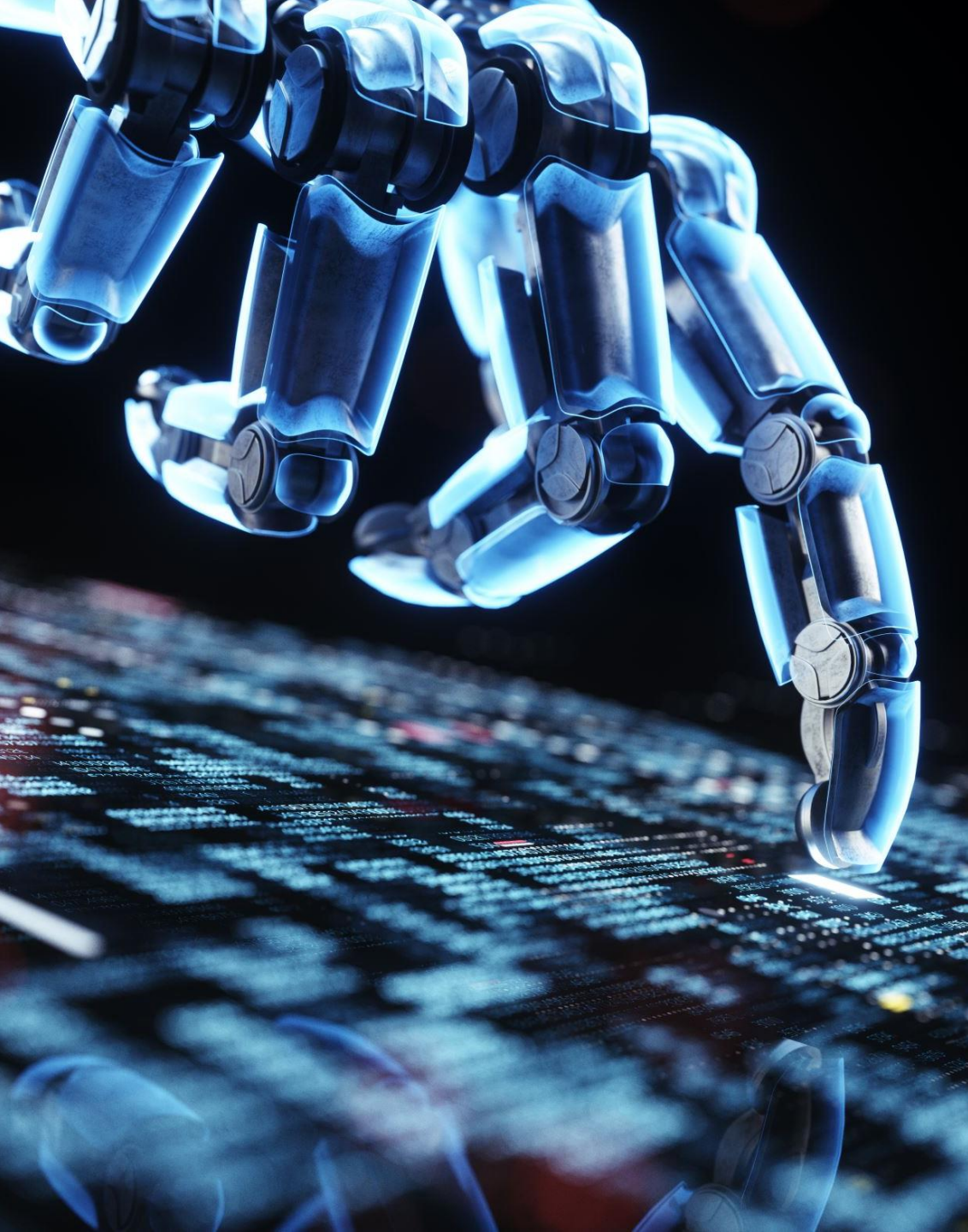
UltraNav

Denis Johnson G00403126

Bachelor of Engineering (Honours) in Software and Electronic
Engineering

Atlantic Technological University

2024/2025



Introduction

What is UltraNav?

UltraNav is a robotic car system built around an ESP32, utilizing ultrasonic sensors to perceive its environment and navigate safely by avoiding obstacles. Users can remotely control the robot and monitor its sensor readings via intuitive web and mobile interfaces.



Key Features

Ultrasonic Obstacle Detection - Uses multiple ultrasonic sensors to detect and avoid obstacles in real time.

Remote Control - Operate the robot from anywhere using a web interface or mobile app.

Visual and Audible Feedback - LED and buzzer provide immediate status and warning signals based on sensor data.

Wi-Fi Connectivity - ESP32 enables wireless communication and live monitoring of sensor data.

Hardware Used

Dc Motors

Ultrasonic sensors

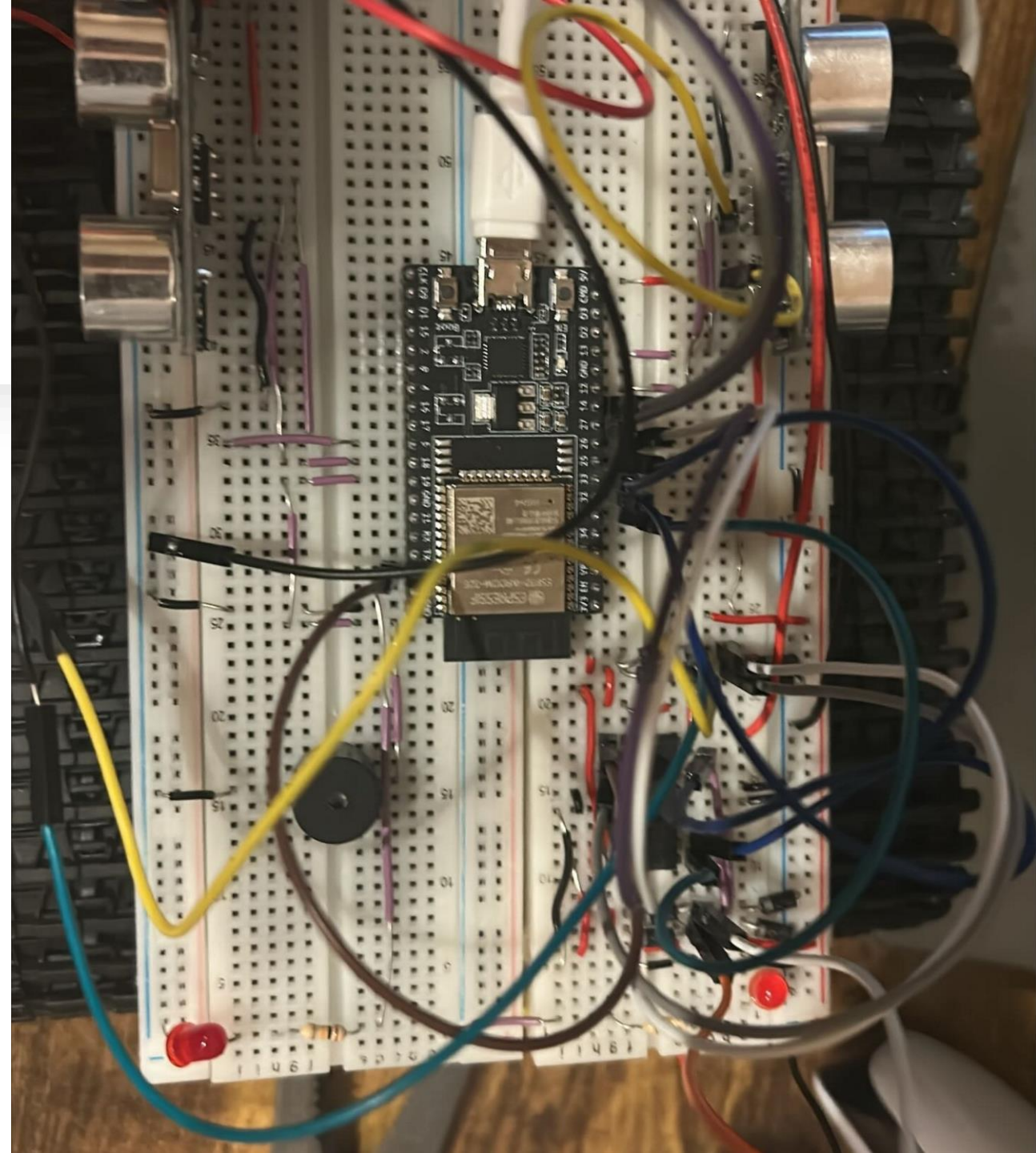
Buzzer

Esp32

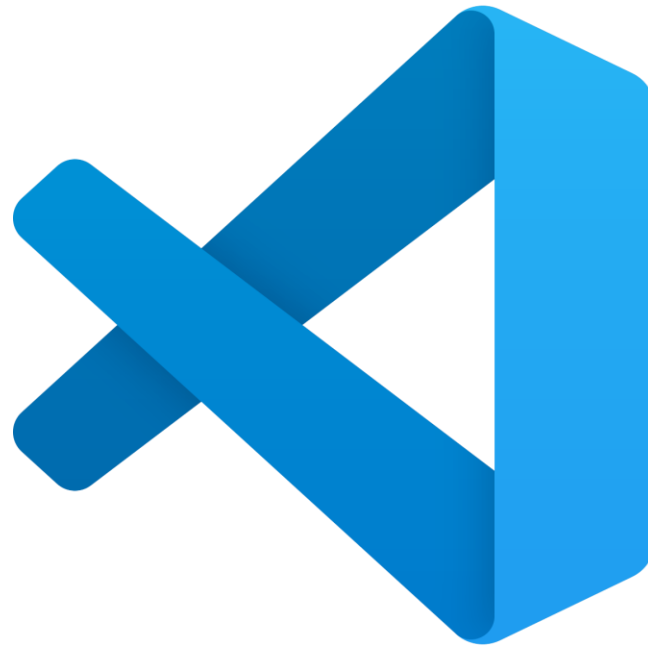
Chassis

9V battery

Motor driver



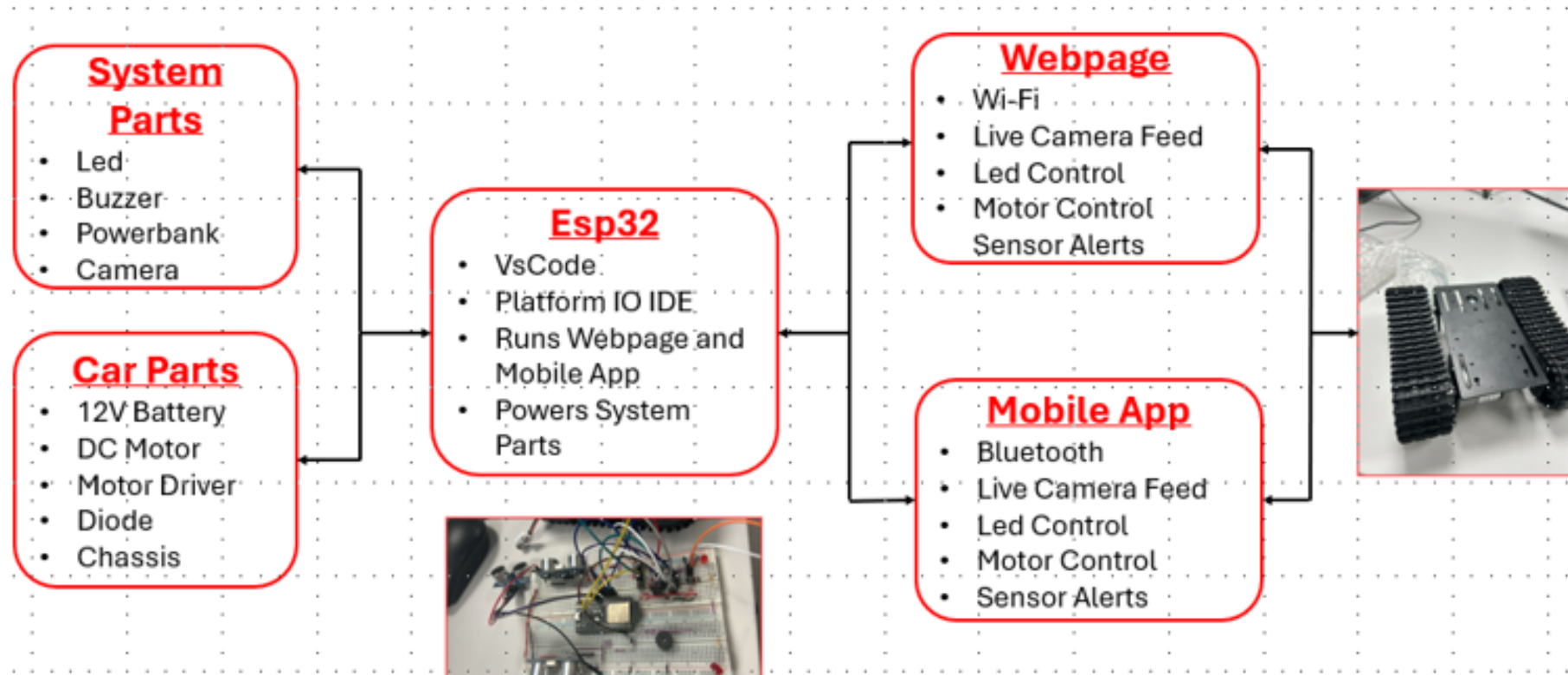
Software Design



VS Code - Used for its advanced code editing, error highlighting, and easy navigation—much more powerful than the Arduino IDE for larger projects.


PlatformIO - Integrated build system and library manager, making it easy to organize, compile, and upload code for the ESP32.

Architecture Diagram




Subsystems Implementation

MOTORS - ENABLE THE ROBOT TO MOVE FORWARD, BACKWARD, AND TURN. CONTROLLED BY THE ESP32 THROUGH A MOTOR DRIVER, RESPONDING TO BOTH USER COMMANDS AND SENSOR INPUT FOR OBSTACLE AVOIDANCE.



ULTRASONIC SENSORS - CONTINUOUSLY MEASURE DISTANCES TO OBSTACLES USING SOUND WAVES. DATA IS PROCESSED BY THE ESP32 TO DETECT HAZARDS AND TRIGGER APPROPRIATE RESPONSES IN OTHER SUBSYSTEMS.

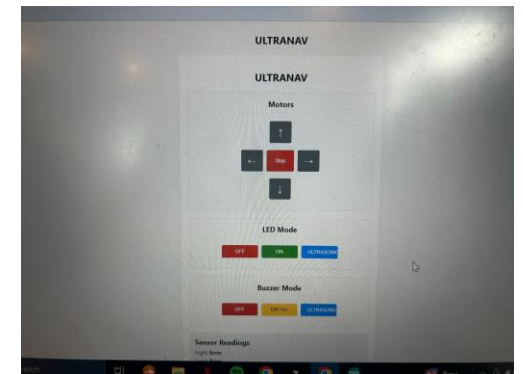
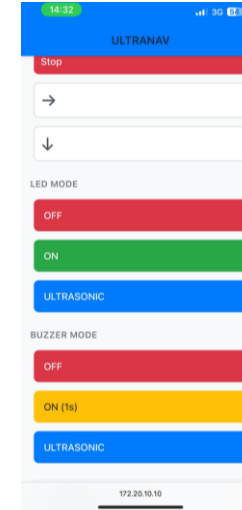
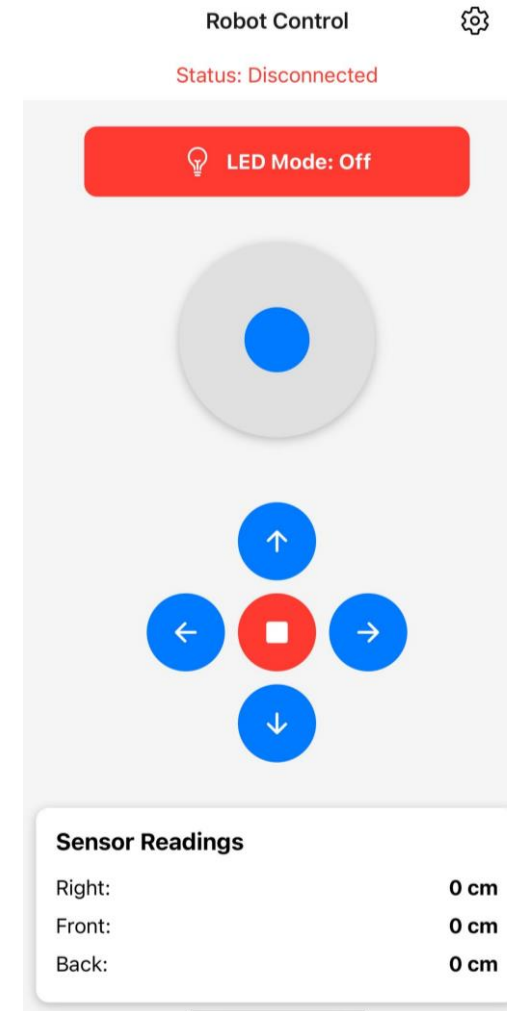


LED & BUZZER - PROVIDE REAL-TIME VISUAL AND AUDIBLE FEEDBACK. THE LED AND BUZZER CHANGE MODES (OFF, ON, ULTRASONIC/WARNING) BASED ON SENSOR READINGS AND USER COMMANDS, KEEPING USERS INFORMED OF THE ROBOT'S STATUS AND ENVIRONMENT.

User Interfaces

Web Interface - Evolved from a simple LED control page to a full-featured dashboard with motor controls, sensor readings, and feedback modes. Allows real-time remote operation and monitoring from any browser.

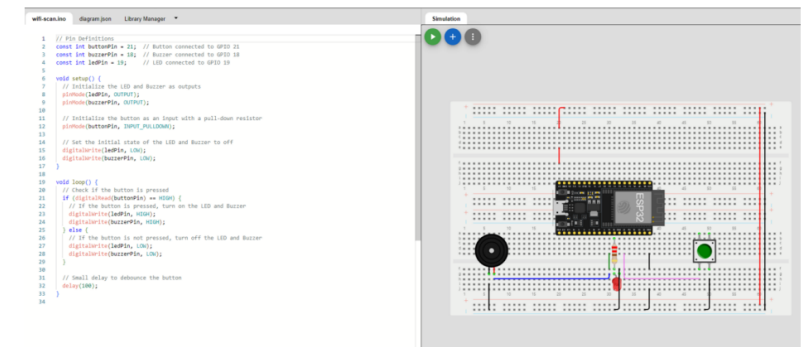
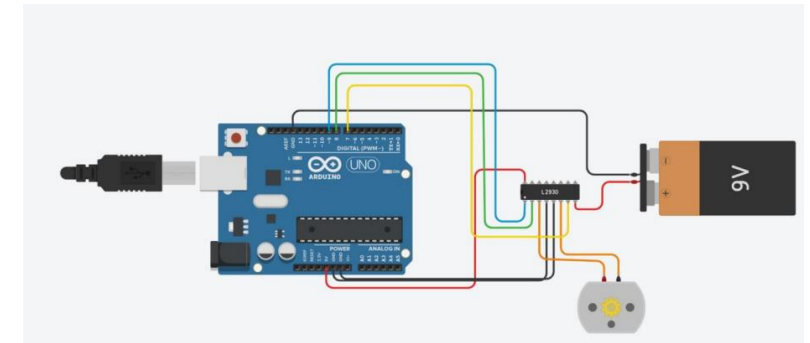
Mobile App - Provides intuitive, touch-friendly controls for movement, LED, and buzzer modes, plus live sensor data. Designed for ease of use and accessibility on smartphones, enhancing user experience and convenience.



Simulation & Testing

Wokwi & Tinkercad - Used both platforms to simulate the ESP32, sensors, and actuators virtually before building the physical circuit.

Benefits - Allowed rapid prototyping, easy debugging, and safe testing of code and wiring minimizing hardware errors and saving time.



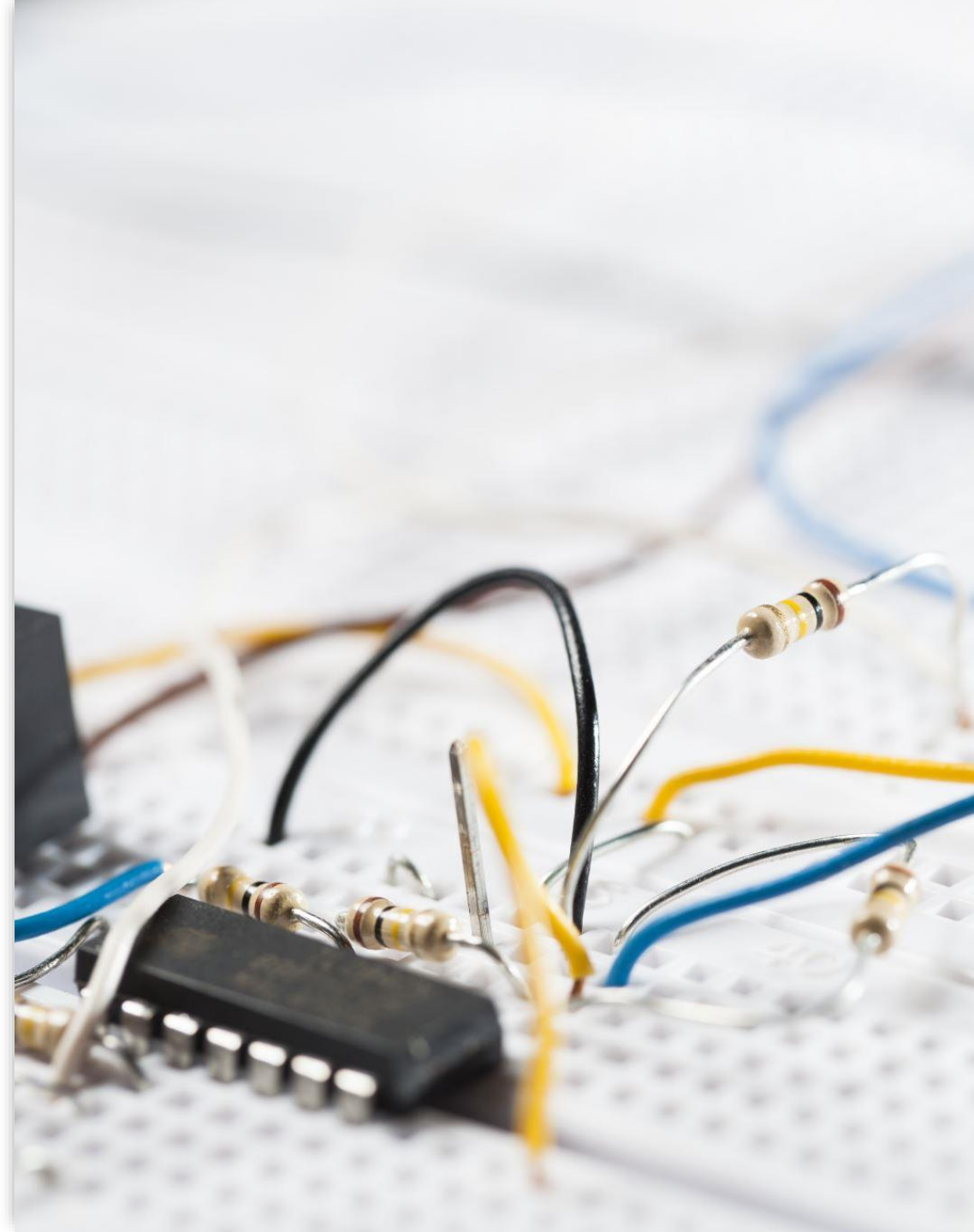
Challenges & Solutions

Wi-Fi Connectivity - Faced issues with stable connections; solved by refining network setup and using serial output for debugging.

Sensor Accuracy - Encountered noisy or inconsistent ultrasonic readings; addressed by filtering data and careful sensor placement.

Pin Conflicts & Power - Some ESP32 pins were input-only or underpowered for motors; resolved by selecting appropriate output pins and using a separate power supply for motors.

User Interface Usability - Early interfaces were basic; improved by iteratively redesigning the web and mobile UI for clarity and ease of use.



Results

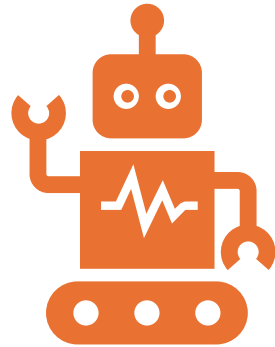


Final System - Successfully built and demonstrated a fully functional ESP32 robot with remote control via web and mobile interfaces.

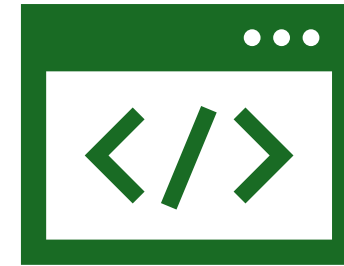
Key Achievements

- Real-time obstacle detection and avoidance
- Responsive visual and audible feedback
- Robust, modular software design
- Reliable Wi-Fi connectivity and user-friendly interfaces

Conclusion



Achievements - Developed a Wi-Fi enabled ESP32 robot with modular hardware and software, real-time obstacle detection, and intuitive web/mobile control.



What I Learned - Gained experience in embedded systems, modular coding, sensor integration, user interface design, and the benefits of simulation and version control.



Thanks For Listening

Thanks for listening if you have any questions, please ask them now

