T.A.L.O.N

(Tactical Animal Locating Optical Navigator)

Team Members:

1. Chittesh D 25BRS1255
2. MOPURU SAI CHARAN REDDY 25BRS1254
3. ANIRUDH SREERAM 25BRS1256
4. KAILASH A 25BEC1065
5. NAYAKPARA PRITHVI MITESH 25BMH1091

What is TALON?

- This project is a smart, telerobotic rover that combines remote operation with real-time, AI-powered object detection.
- This presentation will guide you through the project's mission, design, hardware, software, and future potential.

01

"Can we build a low-cost, smart rover that does more than just drive?"

02

This is a "co-op" system with three main parts that talk over a Wi-Fi network hosted by the rover itself.

Project vision and mission

The Problem:

- Most simple robots are "dumb"—they can't see or understand their environment.
- Their capabilities are limited to basic tasks like following a line or avoiding a wall.

01. Create an intelligent, remote-controlled (telerobotic) system.

02. The robot must stream live video to an operator.

03. A separate AI "brain" must watch this stream and identify specific, pre-determined targets in real-time.

Ideation Process

01

We started with the classic line-follower concept but agreed it was too basic and lacked a "wow" factor for your review.

02

We explored adding more personality or utility, brainstorming ideas like an obstacle-avoiding bot, a "follow-me" pet, or a junction-counting "delivery" robot.

03

The key breakthrough was deciding to use an ESP32-CAM to create a live-streaming FPV (First-Person View) robot, immediately making it a more impressive, modern tech project.

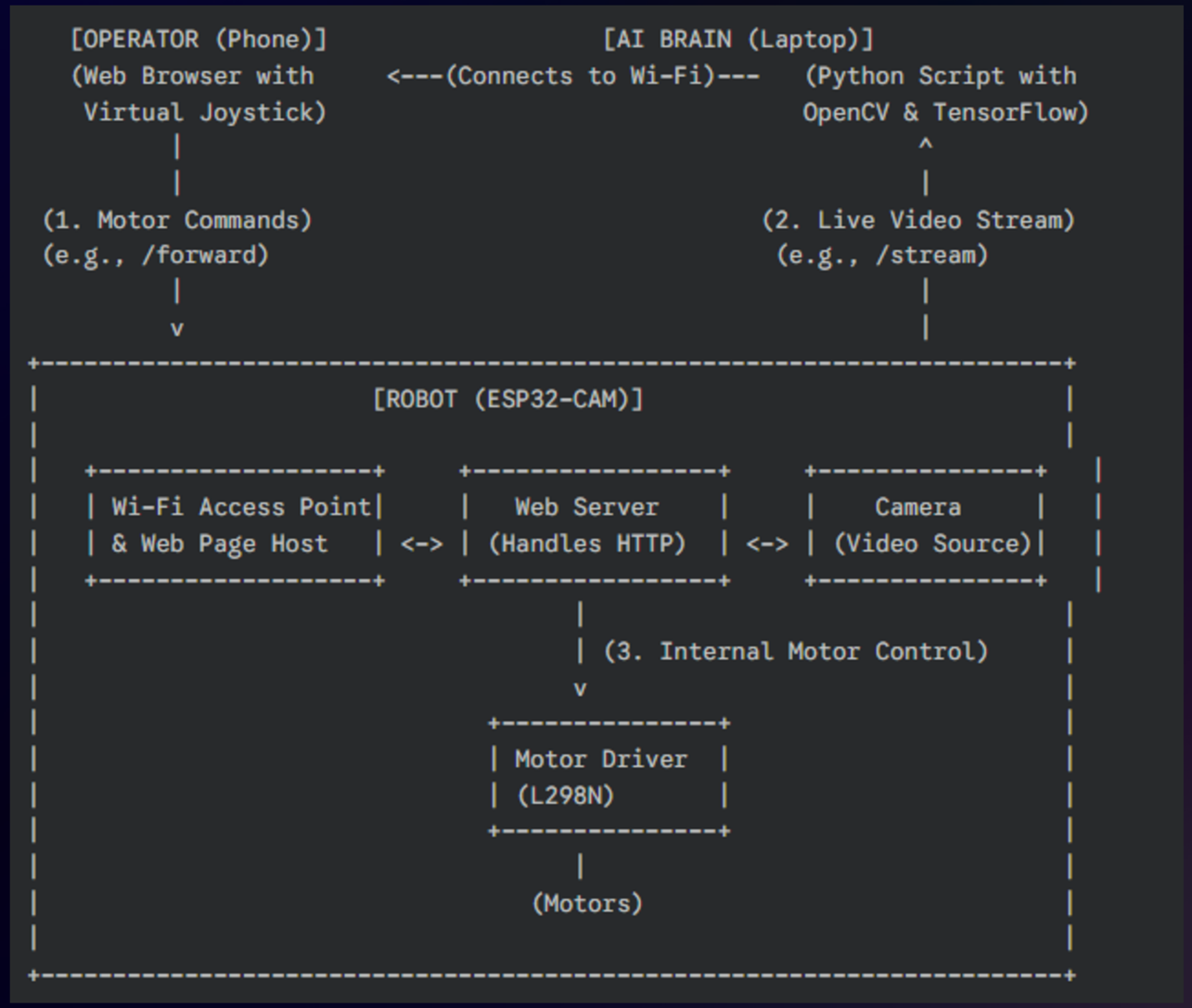
04

We finalized the "T.A.L.O.N." concept by splitting the workload: the robot's only job is to stream video, while a separate laptop runs the complex AI. This made the 1-hour build achievable while resulting in a powerful, interactive AI project.

Hardware

To build the rover itself, you need a 2WD robot chassis kit, which includes the frame, two DC motors, wheels, and a caster. The project's brain and eye is an ESP32-CAM module, which streams video and controls the robot. To drive the motors, you'll use an L298N motor driver module. The power system is split for stability: a 5V USB power bank runs the ESP32-CAM, while a separate 4xAA battery pack (or a 7.4V LiPo) provides power to the L298N for the motors. Finally, a set of jumper wires is needed to connect everything on the robot. For the external tools, you'll need an FTDI programmer (a USB-to-Serial adapter) to upload the code, a laptop to run the Python AI script, and a smartphone to act as the remote controller.

Block Diagram



Final reflections & future steps

This project serves as an excellent platform, and the next logical step is to create a fully autonomous system by enabling the Python AI script to send an HTTP command back to the robot, like /stop or /celebrate, the moment the target is found. To further enhance this, we could move the AI on-board by upgrading the brain to a more powerful board, such as a Raspberry Pi or an ESP32-S3, which would eliminate the need for the laptop entirely. This processing power could also support a "Search and Rescue" mode, where we add a servo and gripper, allowing the robot to autonomously approach and physically interact with the target after finding it. Finally, to improve operator feedback, we could add simple acoustic feedback via a buzzer on the robot that provides an immediate audio cue as soon as the AI acquires the target.

The End

www.reallygreatsite.com

hello@reallygreatsite.com

123 Anywhere Street., Any City