

UNIVERSITY OF BOHOL

Tagbilaran City, Bohol, Philippines

COLLEGE OF ENGINEERING, TECHNOLOGIES,

ARCHITECTURE AND FINE ARTS

COMPUTER ENGINEERING

COMPUTER ENGINEERING DRAFTING AND DESIGN

CPEP223-A

**SUMOBOT AND HOCKEYBOT**

**Submitted by:**

AMILA, CLIFFORD CLINT A.

GUZMAN, EDDEN DAWN MARIE A.

INTRODUCTION

**RATIONALE**

A **Sumobot** is a small autonomous robot designed to compete in a circular ring, where its main objective is to locate and push its opponent out of the boundary, much like traditional Japanese sumo wrestling. These robots are usually built using microcontrollers such as Arduino or Raspberry Pi and are equipped with various sensors, including infrared, ultrasonic, and line sensors. These allow the Sumobot to detect its opponent, avoid going out of bounds, and respond to changes in its environment during battle. Unlike remote-controlled robots, Sumobots are programmed to act independently, making real-time decisions based on sensor input and pre-defined strategies. They are widely used in educational environments to introduce students to important concepts in science, technology, engineering, and mathematics (STEM), including coding, electronics, mechanics, and problem-solving. Sumobot competitions are also common in schools and robotics events, promoting creativity, teamwork, and innovation. In addition to their educational value, they serve as a stepping stone for understanding real-world applications of autonomous systems, such as self-driving cars, automated manufacturing, and artificial intelligence. Through hands-on experience in building and programming Sumobots, learners gain practical skills and a deeper understanding of how intelligent machines interact with their surroundings.

Sumobots come in various designs and sizes, often limited by competition rules that specify weight, dimensions, and power sources. Builders can customize their Sumobots with different chassis materials, wheel configurations, and sensor placements to improve performance and strategy. Some robots are designed to be aggressive, charging straight at opponents with speed and force, while others rely on smarter tactics like waiting for the opponent to make a move or using sensors to dodge and counterattack. Programming plays a crucial role in a Sumobot’s effectiveness, as it determines how the robot reacts in different scenarios whether to advance, retreat, turn, or spin. The process of designing, building, and testing a Sumobot encourages logical thinking, trial and error, and continuous improvement. These projects not only foster a deeper interest in robotics and automation but also prepare students and hobbyists for more advanced challenges in engineering and artificial intelligence. As technology evolves, the principles learned through Sumobot development remain relevant and valuable across a wide range of industries.

A **Hockeybot** is a type of mobile robot designed to simulate the gameplay of ice or table hockey by automatically tracking, aiming, and hitting a puck into a goal. Typically used in robotics competitions and educational projects, Hockeybots are equipped with various components such as wheels for movement, sensors (like infrared or ultrasonic) for puck detection and obstacle avoidance, and actuators or servo motors to control the striker or stick mechanism. These robots are often programmed using platforms like Arduino or Raspberry Pi, allowing them to respond intelligently to the movement of the puck and the opposing player. Hockeybots can be autonomous or remotely controlled, depending on the design and competition rules. Their main functions include locating the puck, navigating the playing field, positioning themselves strategically, and accurately hitting the puck toward the goal. Beyond competition, Hockeybots are excellent tools for learning robotics, physics, mechanical design, and real-time control systems. They provide a fun and interactive way to understand how automation, sensor integration, and algorithmic decision-making work together in dynamic environments, mirroring real-world applications in sports technology, automation, and AI-based control systems.

**Sumobots and Hockeybots** are two popular types of educational and competitive robots designed to simulate real-world scenarios through engaging and interactive challenges. A **Sumobot** mimics sumo wrestling by autonomously detecting and pushing opponents out of a ring using strategically placed sensors and programmed movement. Meanwhile, a **Hockeybot** is built to play a game of hockey, locating and striking a puck toward a goal while navigating around obstacles or opponents. Both robots typically use microcontrollers like Arduino or Raspberry Pi and rely on components such as wheels, motors, sensors, and mechanical frames to perform their tasks. These robots are commonly used in robotics competitions, school projects, and workshops to teach important STEM concepts such as electronics, programming, mechanical design, and problem-solving. By building and programming these bots, students and hobbyists gain hands-on experience with real-time decision-making, automation, and control systems. More than just fun machines, Sumobots and Hockeybots foster creativity, teamwork, and innovation, making them excellent tools for developing future-ready skills in robotics and technology.

**SCOPE AND LIMITATION**

**Sumobot and Hockeybot**

**Scope:**

Sumobots and Hockeybots are widely used in educational, research, and competitive settings to introduce learners to the fundamentals of robotics, automation, and programming. They serve as practical tools for applying concepts in electronics, mechanics, control systems, and artificial intelligence. In schools and robotics clubs, these robots encourage critical thinking, collaboration, and innovation. **Sumobots** are designed for autonomous combat, using sensors and logic to locate and push opponents out of a ring, while **Hockeybots** focus on real-time object tracking and striking a puck toward a goal, simulating fast-paced gameplay. Both robots can be programmed for different strategies and can be customized in terms of design, sensors, and mechanical features, making them flexible platforms for experimentation and learning.

**Limitations:**

Despite their educational value, Sumobots and Hockeybots have several limitations. They often operate in **controlled environments** with clearly defined boundaries, which means their sensors and programming may not perform well in unpredictable or real-world settings. Their **decision-making is limited** by the complexity of their code and the capabilities of their sensors, often relying on basic logic rather than advanced AI. The physical design is also constrained by competition rules such as size, weight, and power limits. Additionally, **Sumobots** may struggle with complex strategies or identifying multiple opponents, while **Hockeybots** can face difficulties in precise aiming and responding to rapid changes in puck position or speed. Power limitations, mechanical wear, and hardware malfunctions can also affect performance. Overall, while excellent for foundational learning, these robots have limited scalability and real-world application outside of educational and recreational purposes.

The **Hockeybot** serves as an effective educational and experimental platform for learning robotics, control systems, and automation. It is commonly used in schools, competitions, and research projects to demonstrate real-time object tracking, motion control, and strategic gameplay. By integrating components such as motors, sensors (like infrared or ultrasonic), and microcontrollers (such as Arduino or Raspberry Pi), the Hockeybot can detect the puck, navigate the playing field, and attempt to score goals. Its scope includes teaching programming logic, mechanical design, and sensor integration, making it a valuable tool for developing problem-solving skills and hands-on technical knowledge. However, the Hockeybot also has several limitations. It operates best in a controlled environment with a smooth, flat surface and clear boundaries. Its accuracy in detecting and striking the puck is limited by sensor range, lighting conditions, and motor precision. Moreover, its response time may lag during fast-paced action, and its programming may not be sophisticated enough to adapt to complex, unpredictable gameplay. Power constraints, limited processing capabilities, and mechanical wear also affect its performance. As such, while the Hockeybot is excellent for educational and experimental purposes, it has limited applicability in real-world dynamic environments beyond the scope of simple games or demos.

**Problem Requirements**

**PURPOSE**

The main **purpose** of both **Sumobot** and **Hockeybot** is to serve as educational tools for teaching and applying fundamental concepts in **robotics, electronics, and programming**. These robots are designed to provide hands-on learning experiences that help students and hobbyists understand how sensors, motors, microcontrollers, and logic work together in real-time systems. The **Sumobot** focuses on autonomous decision-making, object detection, and strategic movement, while the **Hockeybot** emphasizes object tracking, aiming, and motion control. Both are widely used in **robotics competitions** to encourage creativity, teamwork, and problem-solving skills in a fun and challenging environment. Additionally, they help learners develop critical thinking, coding skills, mechanical design abilities, and an understanding of automation—all of which are valuable in modern technology and engineering fields. Beyond education, these robots also promote interest in **STEM careers** and introduce basic principles used in more advanced technologies like self-driving cars, industrial robots, and artificial intelligence systems.

The main objective of both the Sumobot and the Hockeybot is to offer practical and engaging platforms for learning the basics of robotics, programming, and engineering. These robots are crafted to instruct students and enthusiasts on how to merge hardware and software by utilizing sensors, actuators, microcontrollers, and control algorithms. The Sumobot particularly focuses on cultivating skills in autonomous navigation, opponent detection, and strategic decision-making within a competitive setting, whereas the Hockeybot emphasizes object tracking, precision control, and real-time interaction with a moving puck. Both robots enhance problem-solving skills, creativity, and teamwork through hands-on projects and competitions. Beyond technical expertise, these robots promote innovation and iterative design, as builders are required to test and improve their robots for enhanced performance. They also act as motivational instruments to spark interest in science, technology, engineering, and mathematics (STEM) fields by making learning interactive and enjoyable. Furthermore, Sumobots and Hockeybots assist users in grasping significant real-world applications of robotics, such as autonomous vehicles, industrial automation, and robotic sports technologies. Their development encourages critical thinking, adaptability, and the capability to troubleshoot complex systems, which are vital skills in today’s swiftly changing technological environment. In summary, these robots serve not only as educational resources but also as gateways to further exploration in robotics and automation careers.

**OVERALL DESCRIPTION**

**Sumobot and Hockeybot** are two types of educational and competitive robots designed to mimic real-life activities **sumo wrestling** and **hockey**, respectively while providing a hands-on platform for learning robotics, electronics, and programming. A **Sumobot** is an autonomous robot built to detect and push an opponent out of a circular ring using sensors and pre-programmed logic. It focuses on strategy, movement, and opponent detection in a confined arena. On the other hand, a **Hockeybot** is designed to track and strike a puck toward a goal, simulating a hockey game. It requires real-time object detection, precise aiming, and movement control. Both robots use microcontrollers like **Arduino**, along with components such as motors, wheels, and sensors, to function effectively. These projects help students and enthusiasts understand core STEM concepts, including automation, sensor integration, and control systems. They are commonly used in robotics competitions, workshops, and school projects due to their engaging nature and strong educational value. While both robots are limited to controlled environments and simple tasks, they provide an excellent foundation for building more advanced robotic systems in the future.

**SYSTEM REFERENCES**

https://youtu.be/I5G-3mE5Lo8?si=unBzu915KBAWr439

https://youtu.be/eFdeTdUhOfI?si=fz55sZ4OWcJJ\_SVo

https://youtu.be/92J5PbutJGo?si=8MqVKGVjgbZldKB9

https://youtu.be/tfiSxJrISng?si=LMOrCc4Z8QqtMkBE

**Analysis**

**INPUT REQUIREMENTS**

**Sumobot**; A **Sumobot** is a small autonomous robot designed to compete in a Sumo-style match, where the goal is to push the opponent out of a ring (dohyo).

**Key Features:**

* **Autonomous operation** (no remote control).
* Uses **infrared or ultrasonic sensors** to detect the edge of the ring and opponents.
* Usually built with **powerful motors** for pushing.
* Has a **low, wide frame** for stability and to avoid flipping.

**Objectives:**

* Stay inside the ring.
* Detect and push the opponent out of the ring.
* Avoid self-elimination.

**Programming Focus:**

* Obstacle detection.
* Line detection (to avoid going out of the ring).
* Attack and defense strategies.

A **Hockeybot** is a robot designed to play a robotic version of hockey, either autonomously or via remote control, pushing a puck into the opponent’s goal.

**Key Features:**

* Can be **autonomous or manually controlled**.
* Equipped with **puck-detecting sensors** (e.g., color or IR sensors).
* May use a **kicker or striker mechanism**.
* Uses wheels for faster movement and direction control.

**Objectives:**

* Detect and follow the puck.
* Navigate the field.
* Score goals while defending your own.

**Programming Focus:**

* Object (puck) tracking.
* Motion planning and strategy.
* Offensive and defensive behavior.

**Comparison Table**

| **Feature** | **Sumobot** | **Hockeybot** |
| --- | --- | --- |
| Goal | Push opponent out of ring | Score goals with puck |
| Control | Fully autonomous | Autonomous or remote-controlled |
| Sensors | Line, distance, IR | IR, color, camera (optional) |
| Mobility | Fast, stable, torque-focused | Fast, agile, often omni-directional |
| Complexity | Moderate | Higher (esp. with vision or AI) |
| Competition Type | 1v1 or small brackets | Team-based or 1v1 |

**OUTPUT REQUIREMENTS**

The **output requirements** of a **Sumobot** and **Hockeybot** are the specifications related to how the robot performs its tasks. These include its **actuators**, movement behavior, sensors' reactions, and other response mechanisms. Here’s a breakdown for each:

**Sumobot Output**

A **Sumobot** is a type of robot designed for sumo-style competitions, where it tries to push the opponent out of a ring.

**1. Motion/Actuation**

* Must drive forward, backward, and turn (left/right) precisely.
* Quick and responsive movement to avoid being pushed.
* High-torque motors to push opponents.

**2. Motor Control**

* Motors must respond to sensor inputs (e.g., line sensors or distance sensors).
* Controlled speed for attack or retreat.
* Sudden bursts for pushing actions.

**3. Sensors Feedback Actions**

* **IR or Ultrasonic Sensors**: Output must include motor direction changes when detecting an opponent.
* **Line Sensors**: Robot must stop, back up, or turn when detecting the ring boundary (white edge).

**4. LED/Buzzer (Optional)**

* LED indication for status (e.g., ready, detecting opponent, victory).
* Buzzer for match start or end signals (optional).

**5. Power Management**

* Output must support enough power to motors without brownouts.
* Batteries must be regulated to maintain consistent motor power.

A **Hockeybot** is designed to play a robotic version of hockey chasing a puck and trying to shoot it into a goal.

**1. Motion/Actuation**

* Must be able to move in all directions (forward, backward, turn).
* Smooth and controlled movement to approach the puck.

**2. Motor Control**

* Motors must adjust based on puck position.
* Can include kicking/shooting mechanisms (solenoid, servo, or motor-based striker).

**3. Sensors Feedback Actions**

* **IR or Camera (Puck Detection)**: Output includes motor direction changes to follow the puck.
* **Goal Detection**: After scoring, robot stops or changes strategy.

**4. Puck Handling Mechanism**

* Mechanical striker or servo motor to shoot the puck.
* Output includes activation of kicker/striker at proper timing.

**5. LED/Buzzer (Optional)**

* LED indicators for puck detection, shooting, or scoring.
* Buzzer for goal scored or errors (optional).

**6. Power Output**

* Needs regulated power for motors and strikers.
* Ensure striker gets enough voltage/amperage for fast shooting.

**Summary Table**

| **Feature** | **Sumobot** | **Hockeybot** |
| --- | --- | --- |
| Movement | Push, evade, spin | Track puck, shoot, reposition |
| Sensors | IR/Ultrasonic, Line sensors | IR/Camera for puck tracking |
| Actuators | High-torque motors | Motors + Puck kicker (servo/solenoid) |
| Special Output | Push action, boundary avoidance | Puck shooting mechanism |
| Optional Indicators | LED/Buzzer for status | LED/Buzzer for events |

**NECESSARY FORMULA AND THEIR DESCRIPTION**

**N/a**

**Design**

**User Interface Design**

In robotics competitions like **Sumobot** and **Hockeybot**, the **user interface design** plays a crucial role in enhancing control, monitoring, and debugging. Here’s a breakdown of how **UI design applies to both** types of bots:

**1. Sumobot UI Design**

Sumobot is an autonomous robot designed to push its opponent out of a ring. It usually runs without human intervention during matches, but UI is still important in the following contexts:

**Setup and Configuration UI**

* **Purpose:** Configure sensors, test motors, calibrate movement.
* **Design Features:**
  + Sensor status indicators (IR, ultrasonic, line sensors).
  + Motor test buttons (forward, backward, turn).
  + PID tuning sliders (for speed and turning behavior).
  + Battery level indicator.

**Pre-Match Control Panel (Optional)**

* Select strategy mode (aggressive, defensive).
* Start/Stop system test.
* Save/load configuration profiles.

**Post-Match Debugging UI**

* Log viewer (collision logs, sensor triggers).
* Replay path or sensor map of the last match.
* Error reports or fault codes.

**Tools Used:**

* PC/Mobile App (Python GUI, Processing, web-based).
* Simple OLED or LCD displays onboard the robot.
* Serial monitor (via Arduino IDE or custom dashboard).

**2. Hockeybot UI Design**

Hockeybot involves more real-time control, so the UI is more **interactive and dynamic**, often involving human control (e.g., joystick, gamepad, remote app).

**Real-Time Control Interface**

* **Purpose:** Allow manual or semi-automatic control.
* **Design Features:**
  + Joystick/gamepad mapping display.
  + Live camera feed (if FPV or visual aid is used).
  + Motor power and direction indicators.
  + Goal counter or puck possession status.

**Control Modes:**

* Manual (via Bluetooth, WiFi, RF controller).
* Autonomous (UI to toggle modes).
* Hybrid (assisted targeting or puck tracking).

**Performance Metrics UI**

* Speed, battery level, and temperature readings.
* Match timer or countdown.
* Real-time score update.

**Tools Used:**

* Android app (MIT App Inventor, Flutter, or custom).
* PC dashboard (Python, C#, Unity-based).
* Web UI (HTML/JS with ESP32 or Raspberry Pi).

**Summary Comparison Table**

| **Feature/Aspect** |  |  |  |  |  |  |  |  |  |  |  | **Sumobot UI** |  |  |  |  |  |  |  | **Hockeybot UI** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Control Type |  |  |  |  |  |  |  |  |  |  |  | Mostly autonomous |  |  |  |  |  |  |  | Manual / hybrid |
| Main UI Use |  |  |  |  |  |  |  |  |  |  |  | Configuration, debugging |  |  |  |  |  |  |  | Real-time control, gameplay stats |
| Display |  |  |  |  |  |  |  |  |  |  |  | LCD/OLED or PC serial monitor |  |  |  |  |  |  |  | Phone/tablet/PC, gamepad display |
| Input |  |  |  |  |  |  |  |  |  |  |  | Buttons, sliders, mode selectors |  |  |  |  |  |  |  | Joystick, touch screen, buttons |
| Feedback |  |  |  |  |  |  |  |  |  |  |  | Logs, LED indicators |  |  |  |  |  |  |  | Live feedback, HUD, audio cues |

**Features of the Project**

Here are the **user interface features** for both **Sumobot** and **Hockeybot**, categorized by function:

**Configuration Features**

1. **Sensor Calibration Panel**
   * IR/ultrasonic sensor test.
   * Threshold adjustment sliders.
2. **Motor Control Test**
   * Forward/backward/turn test buttons.
   * Speed control sliders.
3. **PID Tuning Interface**
   * Adjust proportional, integral, and derivative values.
4. **Mode Selection**
   * Select from different fight strategies (e.g., defensive, aggressive).
5. **Start/Stop Control**
   * Manual override to begin or stop the bot before match.

**Monitoring & Feedback**

1. **Sensor Readout Display**
   * Live distance or line sensor values.
2. **Battery Level Indicator**
   * Real-time voltage or battery percentage.
3. **Status Indicators**
   * LEDs or screen indicators for "Ready", "Fighting", "Error", etc.
4. **Event Logs**
   * Display log of movements or collision detections.

**Post-Match Analysis**

1. **Match Replay (Optional)**
   * Show a path or movement map.
2. **Error Reports**
   * Fault detection and suggestions (e.g., “Right motor unresponsive”).

**Hockeybot – User Interface Features**

**Control Features**

1. **Joystick/Gamepad Support**
   * Analog stick or D-pad mapping.
2. **Touchscreen Controls (if on mobile)**
   * On-screen buttons for movement/shooting.
3. **Mode Switch**
   * Toggle between Manual / Assisted / Auto modes.

**Visual Feedback**

1. **Live Camera Feed (FPV)**
   * Real-time video stream for aiming/navigation.
2. **Puck Tracker Display**
   * Show puck position if vision system is used.

**Game Status Monitoring**

1. **Scoreboard**
   * Track goals for both teams.
2. **Match Timer**
   * Countdown or stopwatch for game duration.
3. **Power Indicators**
   * Battery voltage, motor status, or overheating alerts.

**Audio/Visual Alerts**

1. Beep/Warn for Low Battery or Goal Events
   * Sound notifications or visual flash indicators.

**Settings and Profiles**

1. Save/Load Configurations
   * Store preferred control sensitivity, speed settings, etc.
2. User Profiles (Optional)
   * Different control layouts or modes per player.

**Summary Table of Key Features**

| **Feature Category** | **Sumobot Features** |  |  | **Hockeybot Features** |
| --- | --- | --- | --- | --- |
| Control | Mode selector, test motors |  |  | Joystick/gamepad, touchscreen controls |
| Sensor Data | Live IR/ultrasonic values |  |  | Camera feed, puck tracker |
| Feedback | Logs, battery, error reports |  |  | Scoreboard, timer, battery, alerts |
| Settings | PID tuning, strategy modes, config profiles |  |  | Control sensitivity, mode switch |
| Visual Display | LCD/PC dashboard |  |  | Mobile/PC app with dynamic HUD |
| Alerts | Error messages, LEDs |  |  | Sound/visual alerts for goals/battery |

**Security and Audit Considerations**

For both Sumobot and Hockeybot, preventing unauthorized access is essential. Access to the robot's configuration interface should be protected with a PIN code or login system, especially when updating firmware or adjusting advanced settings. Physical ports like USB or debug pins should be locked or disabled after initial setup to prevent tampering.

In the case of wireless communication (common with Hockeybots), Bluetooth or Wi-Fi connections must be secured using encryption and secure pairing methods. Communication between controllers and the bot should use encrypted protocols such as TLS or HTTPS to prevent data interception or control hijacking.

Firmware integrity is another key consideration. Firmware updates should be restricted to authorized users, and update processes should include checksum verification or encryption. For additional protection, code signing methods and write-protection features (such as fusing programming pins) can be used to block unauthorized code uploads.

To prevent potential exploits or malfunctions, input rate-limiting should be implemented so that the bot doesn't react to spammed or conflicting commands. Sensors—such as infrared or ultrasonic modules should be shielded and validated to avoid spoofing or interference from external sources like sunlight or IR jammers. Additionally, both types of bots should have built-in failsafe mechanisms that automatically stop motors if invalid data is received or communication is lost.

**Audit Considerations**

From an audit perspective, it’s important to maintain logs of user actions and system behavior. For Sumobots, which are typically autonomous, logging match start and end times, sensor triggers, and motor events helps in post-match analysis and debugging. Hockeybots, which are often human-controlled, benefit from activity logs that capture joystick movements, scoring events, and timing data.

Both bots should log critical system events such as low battery warnings, motor malfunctions, or sensor errors. These logs can be output through a serial monitor, stored on an SD card, or displayed on an onboard screen.

Changes to robot settings such as PID values, control sensitivity, or strategy modes—should also be tracked, especially in team-based or competition settings where multiple users may interact with the same robot. Finally, results of sensor calibration and motor tests should be recorded and time-stamped to ensure accountability and traceability during robot preparation.

**IMPLEMENTATION**

 Url of the saved source code:

https://drive.google.com/drive/folders/12vmYrpkl05DnCqWUuYtWA8x4LCgbQBJq?usp=sharing

Software Package Files and Their Descriptive Information

The software package will include:

Here’s a breakdown of **Software Package Files and Their Descriptive Information** for both **Sumobot** and **Hockeybot** projects. These files are commonly used in robotics projects involving platforms like **Arduino**, **Raspberry Pi**, or **ESP32**, and they support everything from control logic to user interfaces.

| **File Name** | **Description** |
| --- | --- |
| main.ino / sumobot.ino | The main Arduino sketch file containing robot control logic (movement, sensors). |
| hockeybot.ino | Main control code for Hockeybot, including motor control and input handling. |
| config.h | Header file storing constants (e.g., pin mappings, sensor thresholds). |
| motorControl.cpp/h | Custom motor driver file, used to abstract motor logic into reusable functions. |
| sensorManager.cpp/h | Manages all sensor inputs like IR, line sensors, or ultrasonic. |
| bluetoothControl.cpp/h | Handles Bluetooth communication for controller or mobile app integration. |
| ui\_display.cpp/h | Optional file for managing UI components (LCD/OLED). |
| PIDControl.cpp/h | PID control algorithm for smooth motor control and turning (especially in Sumobot). |
| wifi\_server.ino | For Hockeybot with Wi-Fi—serves a web-based control dashboard. |
| app.apk / mobile\_ui.aia | Android control app for Hockeybot, created via MIT App Inventor or Flutter. |
| README.md | Describes the project, setup instructions, and usage guidelines. |
| LICENSE | Software license file (MIT, GPL, etc.) stating usage rights. |
| package.json | For web/mobile dashboards (Node.js), contains dependencies and metadata. |
| data\_log.csv | Optional file storing match or test logs for audits or debugging. |

**FUNCTION DECLARTIONS AND THEIR DESCRIPTITVE PURPOSES**

**1. .ino files (e.g., main.ino, sumobot.ino)**

These are the main Arduino program files that define the behavior of the robot. They include setup routines, loop functions, and calls to sensors and motors.

**2. .cpp and .h files (e.g., motorControl.cpp)**

These are source and header files for modular code. They separate different components like motor logic, sensor management, or communication into reusable modules. This improves code organization and maintainability.

**3. config.h**

Stores all customizable values such as sensor thresholds, robot speed, PID constants, and pin configurations. It helps users tune the robot without digging into the main logic.

**4. PIDControl.cpp/h**

Implements PID (Proportional-Integral-Derivative) control algorithms used in precise motor control, especially useful for maintaining direction or balance in autonomous bots.

**5. wifi\_server.ino or bluetoothControl.cpp**

Handles wireless communication with external devices such as smartphones or web apps. Used more in Hockeybot for real-time control.

**6. app.apk / mobile\_ui.aia**

Mobile apps created with MIT App Inventor or other platforms. These provide a graphical interface for controlling the bot via Bluetooth or Wi-Fi.

**7. README.md**

A markdown documentation file typically found on GitHub. It explains the purpose of the project, how to install it, and how to use it. Great for users and collaborators.

**8. LICENSE**

Defines the legal terms under which the software can be used, modified, or distributed. Common licenses include MIT, GPL, and Apache.

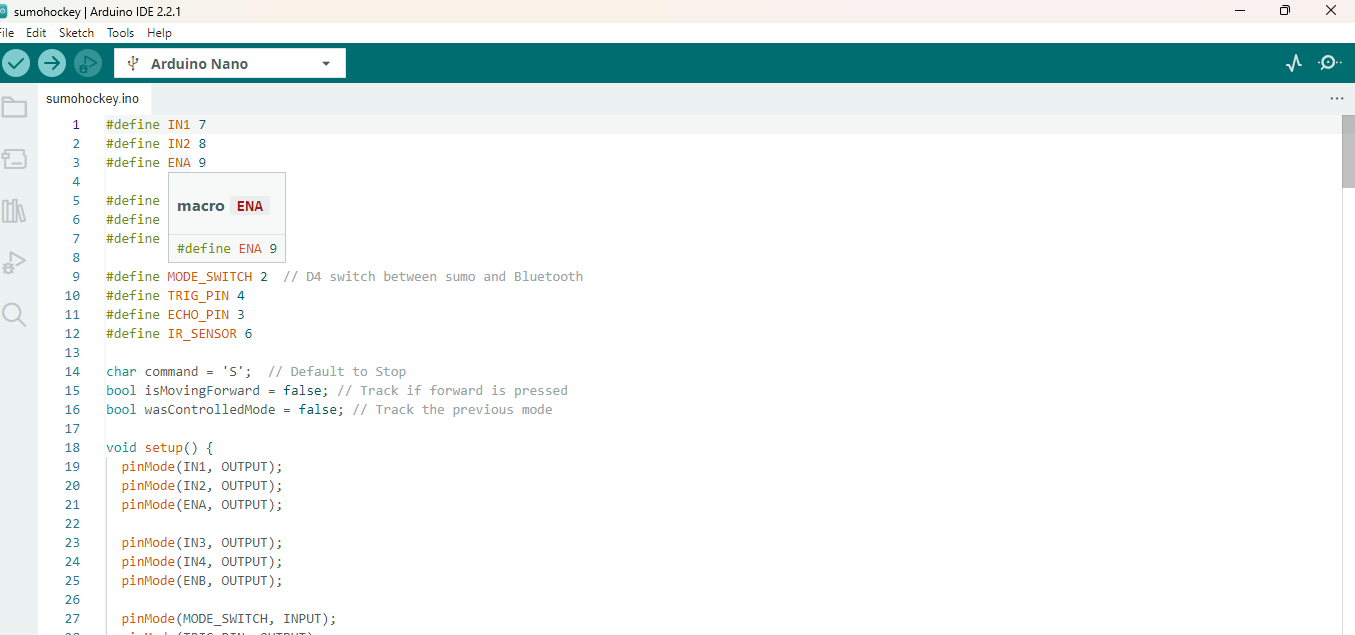
**9. data\_log.csv**

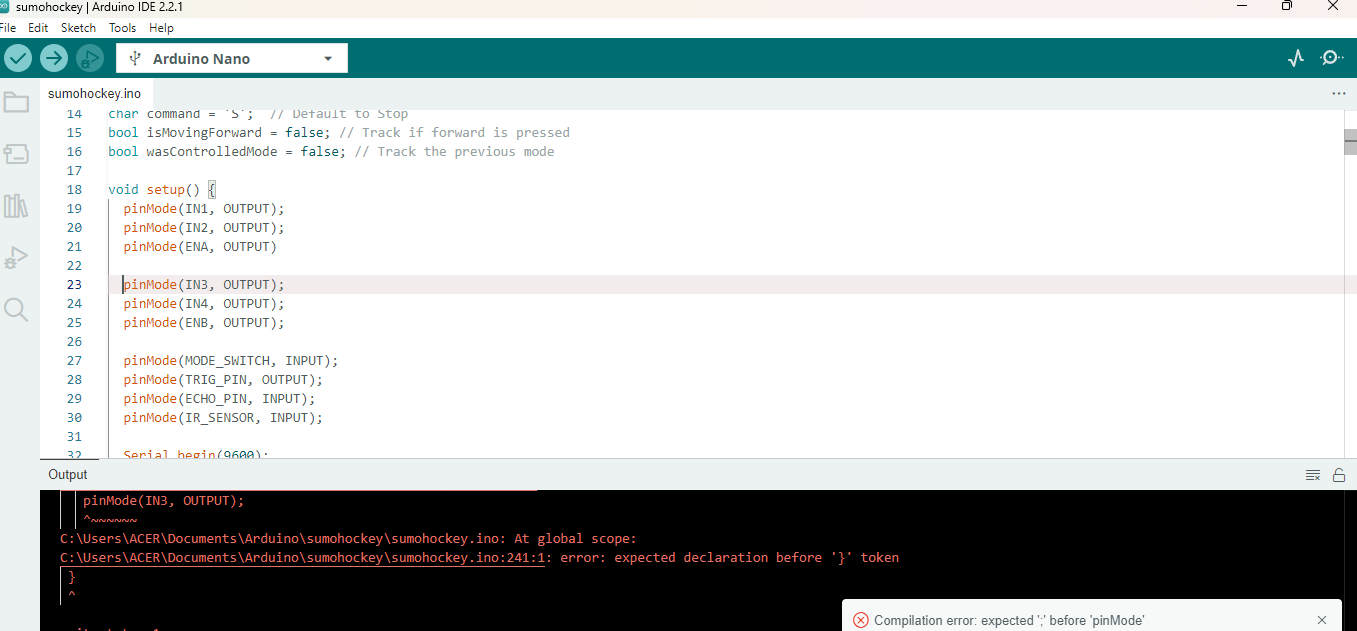
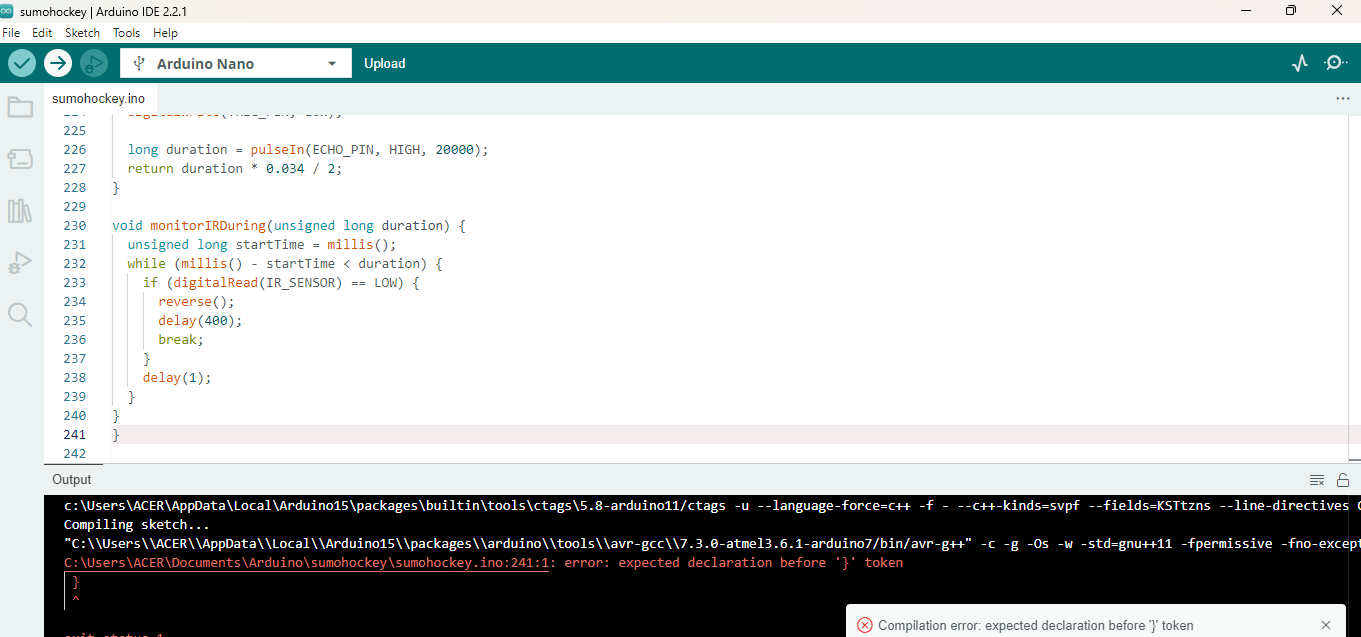
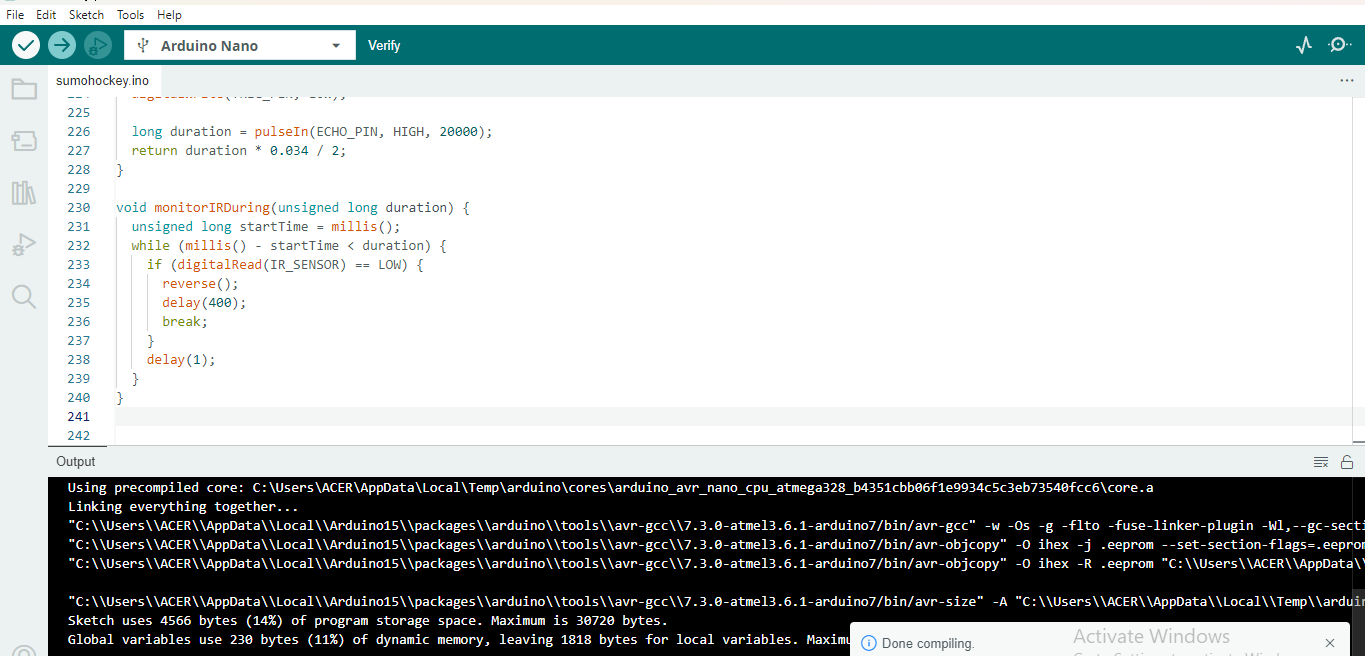
A file where runtime data such as sensor readings, errors, and control events are stored for later analysis. Useful for audits, debugging, or improving performance.

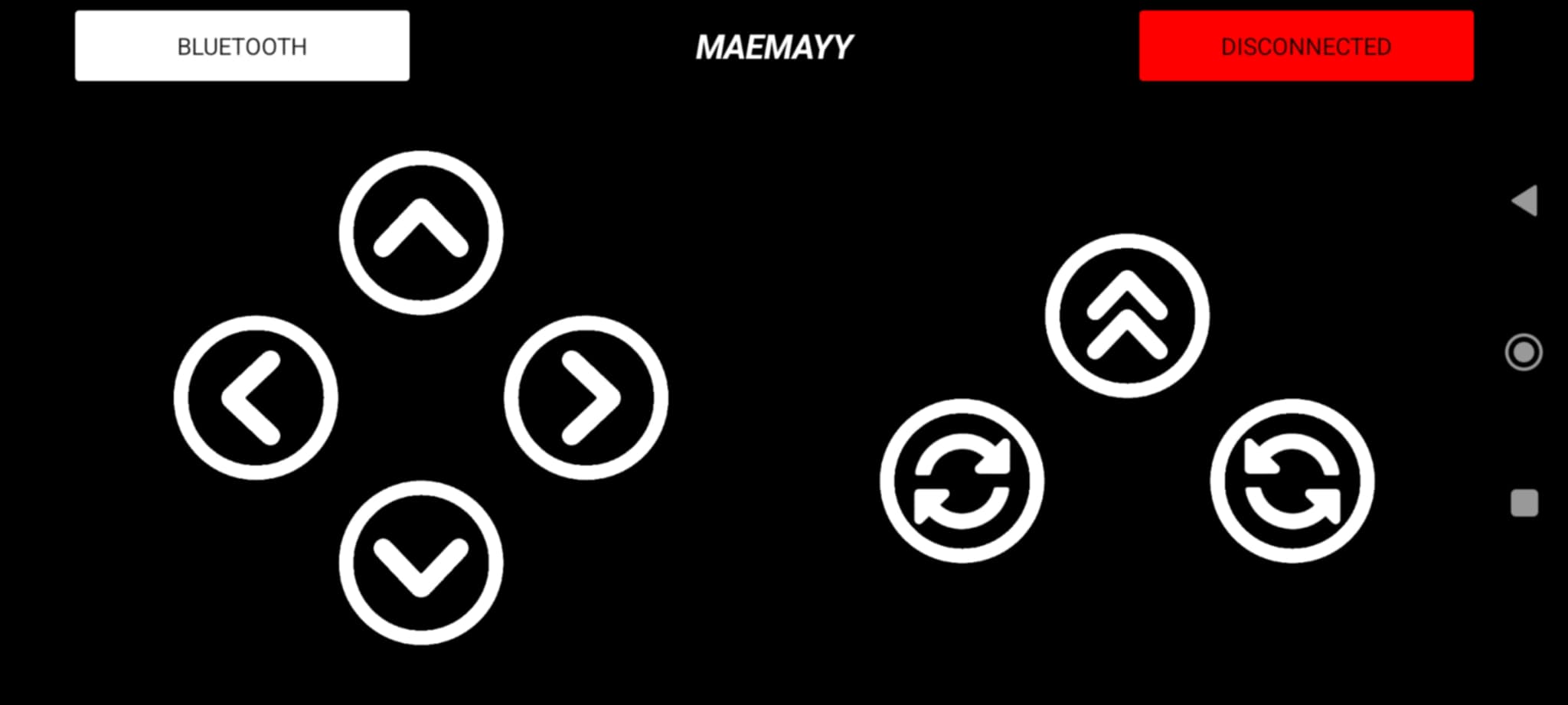
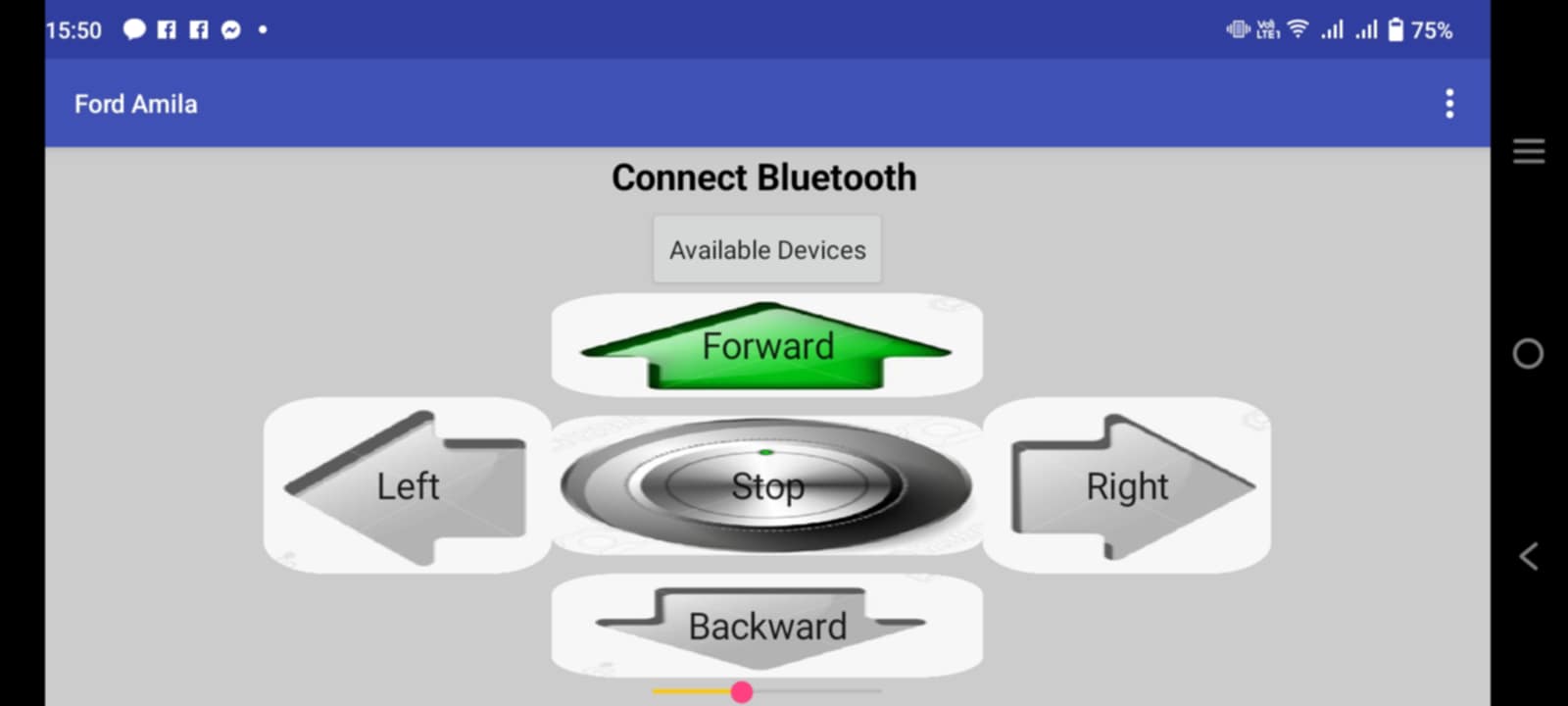
**10. package.json**

If your project uses Node.js (e.g., for web dashboards), this file lists dependencies, scripts, and metadata for your project.

**TESTING AND DEBUGGING**

**Sample Run**



****

**The Team**

# Names and Tasks

Amila, Clifford Clint A.

* Project Developer of Overseeing project development and coordination
* Writer of Designing the user interface and user experience

Guzman, Edden Dawn Marie A.

* Project Developer of Overseeing project development and coordination
* Writer of Designing the user interface and user experience

CURRICULUM VITAE

I. PERSONAL INFORMATIONS

NAME : GUZMAN, EDDEN DAWN MARIE A.

COURSE : COMPUTER ENGINEERING

BIRTHDATE : OCTOBER 17, 2004

BIRTHPLACE : GMP HOSPITAL TALIBON, BOHOL

NATIONALITY : FILIPINO

RELIGION : ROMAN CATHOLIC

GENDER : FEMALE

ADDRESS : PUROK 5-A SUNNYBROOK POBALCION, ALICIA, BOHOL

EMAIL : EDMAGUZMAN@UNIVERSITYOFBOHOL.EDU.PH

II. EDUCATIONAL BACKGROUND

TERITARY: UNIVERSITY OF BOHOL, COLLEGE OF ENGINEERING,

TECHNOLOGY, ARCHITECTURE AND FINE ARTS

DR. CECILIO PUTONG STREET, COGON TAGBILARAN CITY, BOHOL

2023 – PRESENT

SECONDARY: ALICIA TECHNICAL VOCATIONAL HIGH SCHOOL &

MONTESSORI EDUCATIONAL LEARNING CENTRE OF UBAY INC.

S.Y. 2017 – 2023

ELEMENTARY: ALICIA CENTRAL ELEMENTARY SCHOOL

S.Y. 2011 – 2017

NAME : AMILA, CLIFFORD CLINT A.

COURSE : COMPUTER ENGINEERING

BIRTHDATE : DECEMBER 08, 2004

BIRTHPLACE : CEBU CITY, CEBU

NATIONALITY : FILIPINO

RELIGION : ROMAN CATHOLIC

GENDER : MALE

ADDRESS : CANSUNGAY, PILAR, BOHOL

EMAIL : [CCAAMILA@UNIVERSITYOFBOHOL.EDU.PH](mailto:CCAAMILA@UNIVERSITYOFBOHOL.EDU.PH)

II. EDUCATIONAL BACKGROUND

TERITARY: UNIVERSITY OF BOHOL, COLLEGE OF ENGINEERING,

TECHNOLOGY, ARCHITECTURE AND FINE ARTS

DR. CECILIO PUTONG STREET, COGON TAGBILARAN CITY, BOHOL

2023 – PRESENT

SECONDARY: PILAR TECHNICAL VOCATIONAL HIGH SCHOOL

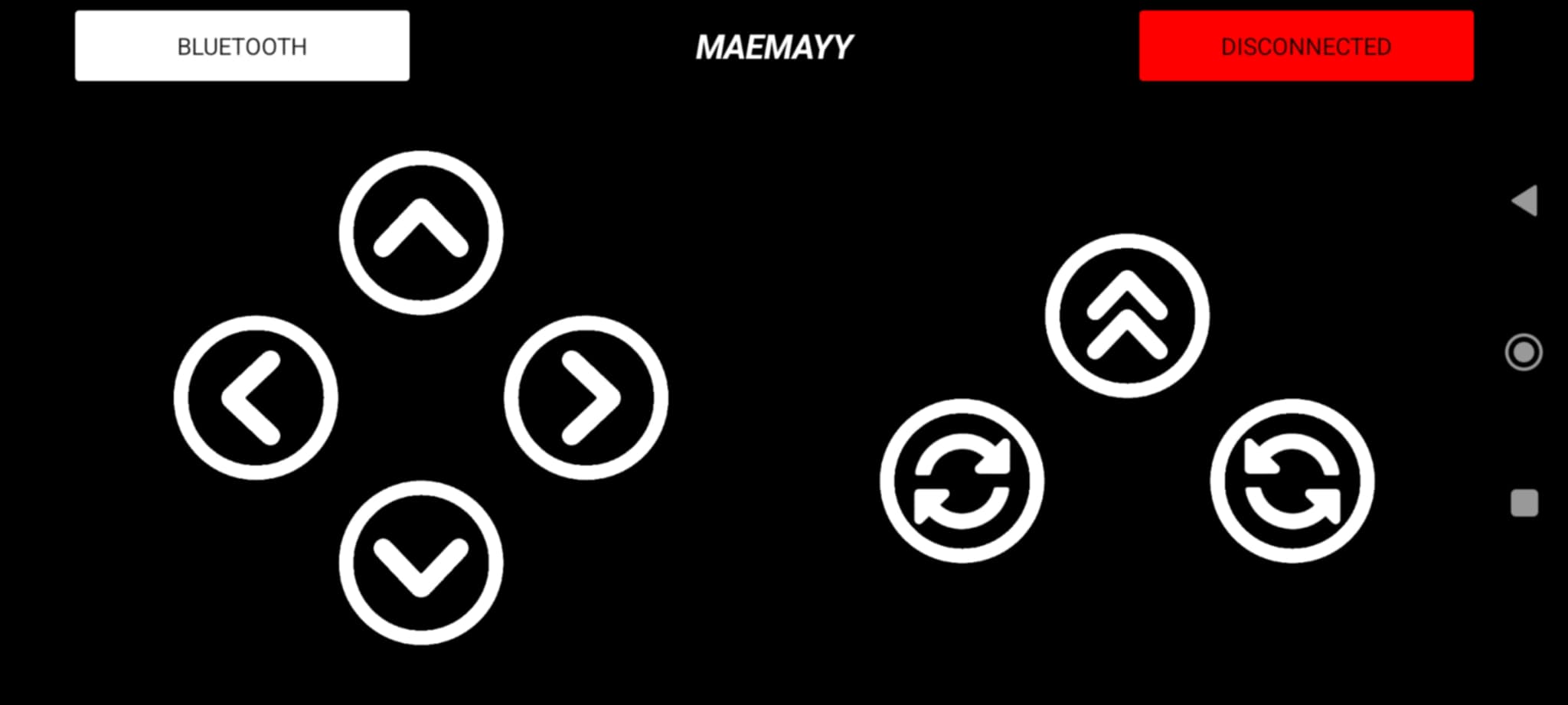
S.Y. 2017 – 2023

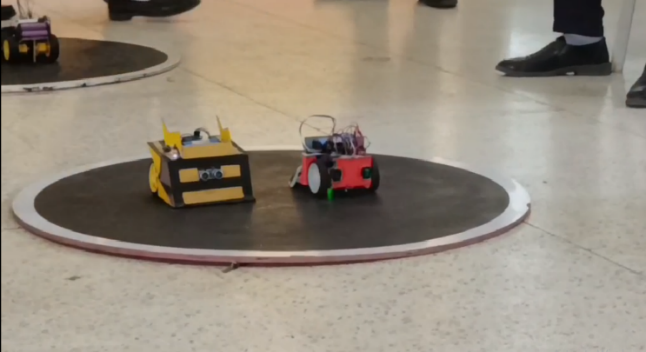
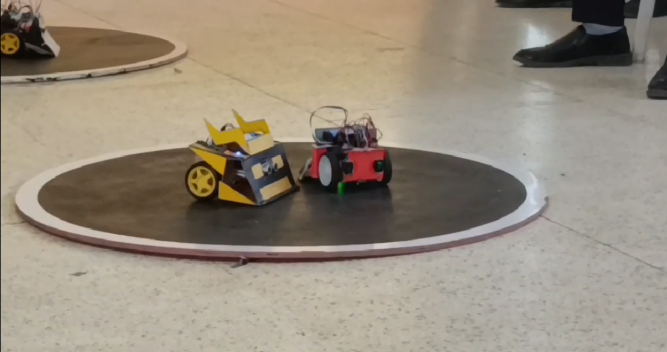
ELEMENTARY: CANSUNGAY ELEMENTARY SCHOOL

S.Y. 2011 – 2017

**Documentation**

****

****

****

**FUTURE DEVELOPMENT**

The **future development** of **Sumobot** and **Hockeybot** is expected to focus on greater intelligence, adaptability, and real-world applicability. As technology advances, these robots can be upgraded with more advanced sensors such as **computer vision cameras**, **machine learning algorithms**, and **AI-based decision-making systems** to enable smarter, more reactive behavior. Future Sumobots may be able to analyze opponent movement patterns, adjust tactics in real time, and even learn from past matches to improve performance. Similarly, Hockeybots could benefit from enhanced object recognition, better motion prediction, and improved accuracy in puck control and goal-scoring.

Wireless communication and **team-based programming** may also become more common, allowing robots to work together in coordinated strategies. Additionally, future designs will likely include more durable materials, compact power systems, and modular components for easier upgrades and repairs. These developments will not only increase the competitiveness of robotics events but also bring educational robotics closer to real-world applications such as **autonomous vehicles**, **sports analytics**, **robotic automation**, and **AI-driven machines**. As students and developers continue to push the boundaries of what these robots can do, Sumobot and Hockeybot will evolve into even more powerful platforms for innovation, research, and skill development in the rapidly growing field of robotics.

**PROJECT COST**

This project cost was as follow;

1. Arduino (Nano) - 136.00

2.Ultrasonic - 140. 00

3. Lead - 50. 00

4. IR sensor - 50.00

5. Bluetooth module - 260.00

6. Wirings - 50.00

7. Motor driver - 60.00

8. Wheels - 118.00

9. Battery - 108.00

10. Battery holder - 50.00

11. Chasis - 100.00

12. Breadboard - 100.00

13. Switch - 30.00

14. Geared Motor - 118.00

**Glossary**

**Autonomous** - responding, reacting, or developing independently of the whole.

**Robotics** - Robotics is a branch of engineering and computer science that involves the conception, design, manufacture and operation of [robots](https://www.techtarget.com/searchenterpriseai/definition/robot). The objective of the robotics field is to create intelligent machines that can assist humans in a variety of ways.

**Electronics** - is the branch of science that deals with the study of flow and control of electrons (*electricity*) and the study of their behavior and effects in vacuums, gases, and semiconductors, and with devices using such electrons.

**Hockey -** is a term used to denote a family of various types of both summer and winter team sports which originated on either an outdoor field, sheet of ice, or dry floor such as in a gymnasium. While these sports vary in specific rules, numbers of players, apparel, and playing surface, they share broad characteristics of two opposing teams using sticks to propel a ball or disk into a goal.

**Hockeybot**- In the realm of robotics, where innovation meets excitement, robo-hockey competitions have emerged as a captivating spectacle of technology, strategy, and teamwork.

**Programming -**is, quite literally, all around us. From the take-out we order, to the movies we stream, code enables everyday actions in our lives. Tech companies are no longer recognizable as just software companies — instead, they bring food to our door, help us get a taxi, influence outcomes in presidential elections, or act as a personal trainer.

**Sumobot** -is an [engineering](https://en.wikipedia.org/wiki/Engineering) and [robotics](https://en.wikipedia.org/wiki/Robotics) competition in which two [robots](https://en.wikipedia.org/wiki/Robot) attempt to push each other out of a circular arena, in a similar fashion to the sport of [sumo](https://en.wikipedia.org/wiki/Sumo). The robots used in this competition are called "sumo robots", "sumobots" or simply "sumos.

**Bibliography**

Wikipedia contributors. (2025, May 9). *Robot-Sumo*. Wikipedia. <https://en.wikipedia.org/wiki/Robot-sumo>

*technoxian*. (2023, July 20). TechnoXian. https://www.technoxian.com/robo-hockey

autonomous. (2025). In *Merriam-Webster Dictionary*. <https://www.merriam> webster.com/dictionary/autonomous

Yasar, K., & Hanna, K. T. (2024, September 4). *What is Robotics?* WhatIs. https://www.techtarget.com/whatis/definition/robotics

Das, S. (2024, September 4). *Electronics Definition & meaning in simple words*. Electronics Tutorial | Best Electronics Tutorial Website. https://www.electronicsandyou.com/electronics-definition.html

Codecademy. (n.d.). What is Programming? And How to Get Started. Codecademy. https://www.codecademy.com/article/what-is-programming