RISC V Product Development Hackathon

Stage 1 - Product Idea submission form

1. Product title → "ArtWheels"

2. *What does your product do?*

- *Draw*
- *Move*
- Turn
- *Adjust*
- Communicate

3. What all interfaces of the board *will be used* in the product ?

- *GPIO pins for Motor Control*
- *GPIO pins for Ultrasonic sensor*
- Serial Communication pins for Bluetooth Module

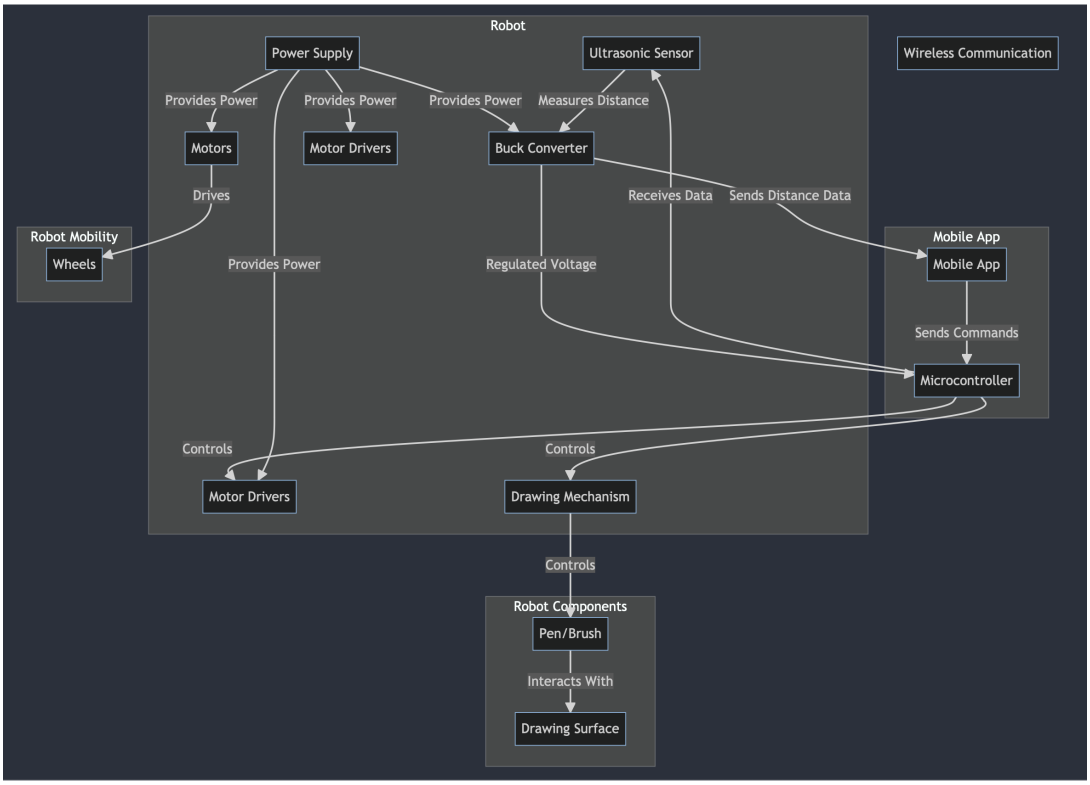
4. Does the product utilise sensors?

- *Yes*

5. If "Yes" for above question, then list your sensors here

- *Ultrasonic sensor (HC-SR04)*

6. Draw a Block diagram of the product.



Block diagram and flow of the working overview of the product

---

7. Upload the Algorithm flowchart of the product.

https://drive.google.com/file/d/13_mvRX_st0DrLiZuJ2OglAk_QtGYTKGK/view?usp=sharing

ArtWheels Algorithm Flowchart:

1. **Start**:
   - Initialisation of the robot and mobile app.
2. **Receive Command**:
   - Continuously listen for commands from the mobile app.
3. **Command Analysis**:
   - Analyse the received command:
     - If "Draw" command:

        - Activate drawing mechanism.
        - Ensure contact with the drawing surface.
    - If "Move" command:
        - Implement motor control logic for movement.
    - If "Turn" command:
        - Adjust motor speeds for turning.
    - If "Adjust" command:
        - Execute adjustments based on the command.

4. **Distance Sensing**:
    - Continuously measure the distance to the nearest obstacle using the ultrasonic sensor.
5. **Obstacle Detection**:
    - Check the distance data:
        - If an obstacle is detected within a predefined range:
            - Stop the robot's movement or adjust its path to avoid collisions.
6. **Drawing Control**:
    - Ensure the drawing mechanism is engaged when in "Draw" mode.
7. **Movement Control**:
    - Implement motor control logic for movement when in "Move" mode.
8. **Turning Control**:
    - Adjust motor speeds for turning when in "Turn" mode.
9. **Feedback to Mobile App**:
    - Send real-time feedback to the mobile app, including distance data and robot status.
10. **Continuous Monitoring**:
    - Continuously monitor the status of the robot, sensors, and drawing mechanism.
11. **Safety Measures**:
    - Implement safety mechanisms to handle exceptional situations (e.g., emergency stops).

12. **End**:
    - End of the algorithm.

---

8. Explain the algorithm of the product in bullet points.

> ### ArtWheels Algorithm:

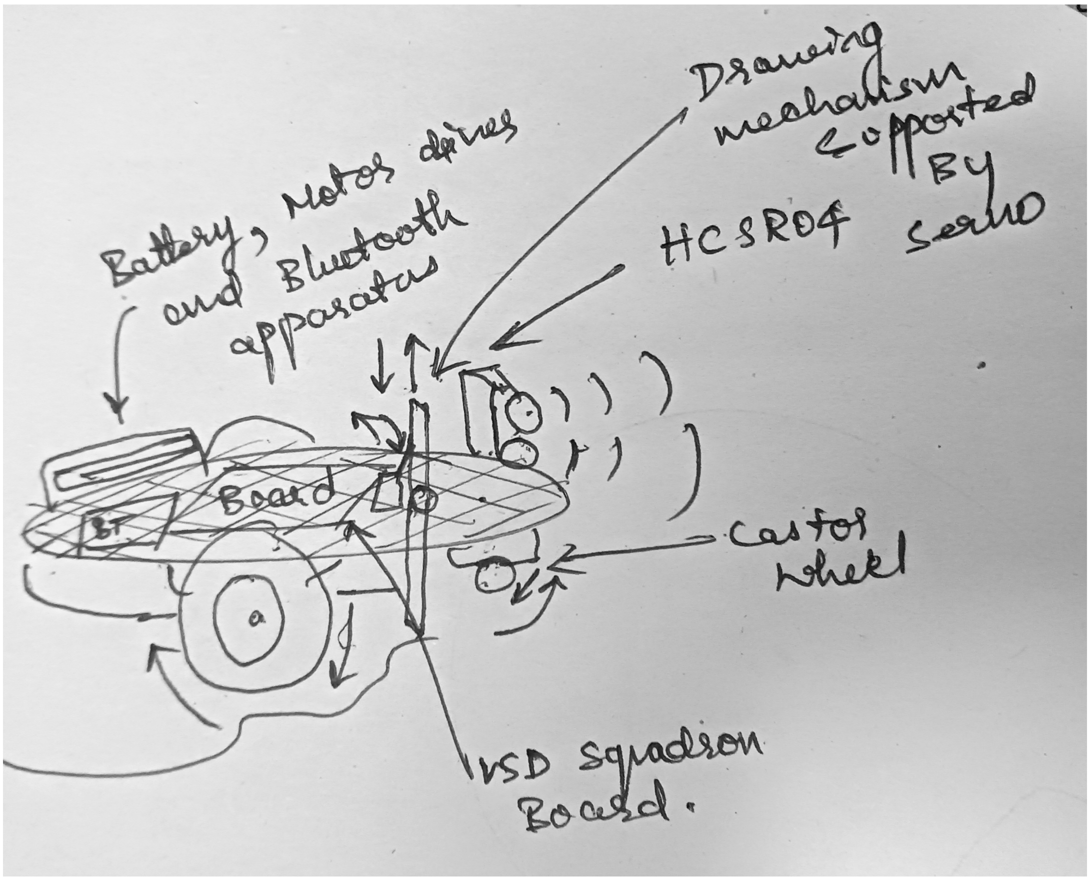
- **Initialisation:**
    - Initialise pins for servo, ultrasonic sensor, and motor control.
    - Define global variables and constants for the project.
    - Set the servo to its neutral position.
- **Main Loop:**
    - Continuously check for incoming Bluetooth commands.
    - If a Bluetooth command is received, process it accordingly.
- **Processing Bluetooth Commands:**
    - Commands include 'U' (Raise drawing mechanism), 'C' (Lower drawing mechanism), 'B' (Reverse movement), 'F' (Set custom motor speeds for forward motion), 'N' (Stop motors), 'J' (Switch to joystick control mode), 'S' (Switch to sensor control mode), and 'A' (Adjust servo angle).
    - Based on the command, perform the following actions:
        - Adjust the servo angle if needed.
        - Control motor speeds for movement.
        - Switch between joystick and sensor control modes.
        - Send or receive sensor data via Bluetooth.
- **Measuring Distance:**
    - Use the ultrasonic sensor to measure the distance to obstacles.
    - Trigger the sensor and calculate the distance based on the pulse duration.
    - Return the distance value.
- **Sending Sensor Data:**
    - Send sensor data (distance) to the connected app via Bluetooth.

    - Prefix the data with an identifier ('D') for sensor data.
- **Motor Control:**
    - Control the motors based on the specified control mode (joystick or mobile sensor).
    - Ensure that motor speed values are within the valid range.
    - Map joystick control values to motor speed if needed.
- **Servo Control:**
    - Adjust the servo angle based on commands or user input.
    - Ensure that the new angle is within a valid range (0 to 180 degrees).
    - Use a custom function (`servo_write`) to set the servo angle.
- **Error Handling:**
    - Implement error handling to handle invalid input values or out-of-range angles.
    - Stop motors if invalid motor speed values are received.
- **Custom Servo Function:**
    - Create a custom function (`servo_write`) to set the servo angle.
    - Calculate the pulse duration based on the desired angle.
    - Send the appropriate signals to the servo to achieve the desired angle.

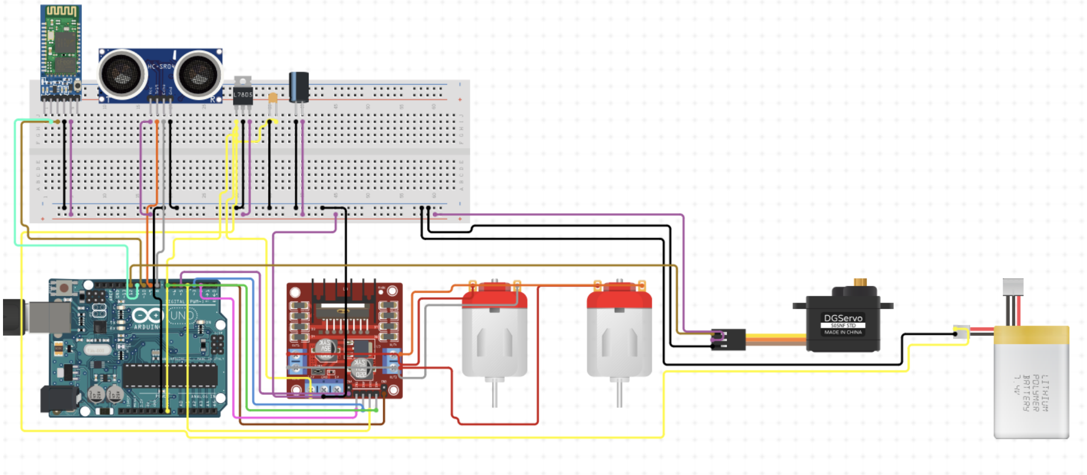
This algorithm controls the movement and servo angle of the RISC V VSD Squadron-based robot using Bluetooth commands and sensor data, allowing for interactive art and control modes.

---

9. Rough sketch of the final product.


Drawing mechanism supported By servo
HC SR04
Battery, Motor drives and Bluetooth apparatus
Board
BT
Castor wheel
VSD squadron Board.

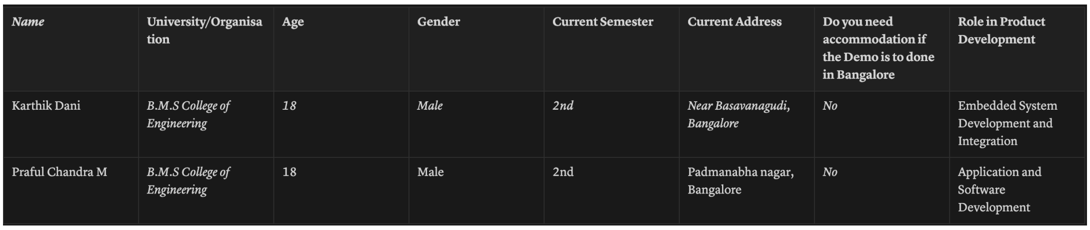
10. Upload the rough sketch of the Internal product (With all connection of components with the board and the product)



11. BoM list (excluding the board) with cost.

Component name	Quantity Required	Unit price (in Rupees)	**Total Price (in Rupees) (Unit price*Quantity)**
HC-SR04 Ultrasonic sensor Reference link	1	61	61 * 1 = 61
2WD Mini Round Double-Deck Smart Robot Car Chassis DIY Kit Reference link	1	414	414 * 1 = 414
TowerPro SG90 Servo Motor Reference link	1	147	147 * 1 = 147
L293D Motor Driver module Reference link	1	82	82 * 1 = 82
HC05 Bluetooth Module Reference link	1	242	242 * 1 = 242
Samsung Icr18650-26Jm 2600Mah (2C) Li-Ion Battery Reference link	2	2 * 353	353 * 2 = 706
18650 Lithium Battery Shield Reference link	1	569	1 * 569
100 uF 50V Capacitor for Smooth Servo control Reference link	1	11	1 * 11 = 11

12. Team details



Name	University/Organisation	Age	Gender	Current Semester	Current Address	Do you need accommodation if the Demo is to done in Bangalore	Role in Product Development
Karthik Dani	*B.M.S College of Engineering*	*18*	*Male*	*2nd*	*Near Basavanagudi, Bangalore*	*No*	Embedded System Development and Integration
Praful Chandra M	*B.M.S College of Engineering*	18	Male	2nd	Padmanabha nagar, Bangalore	*No*	Application and Software Development