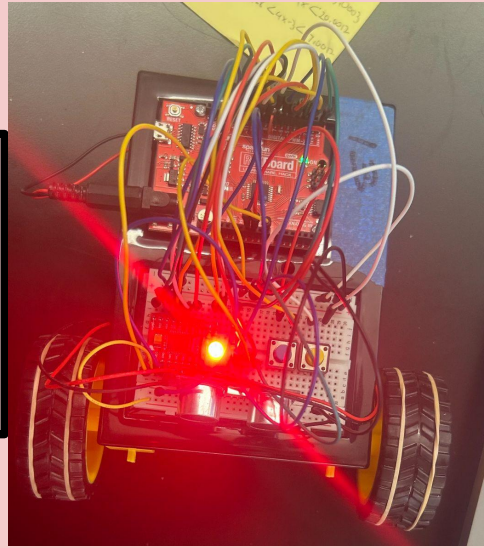
Autonomous Robot



By: Saksham Lubana

Computer Science and Programming, Period 0

Introduction to the project.

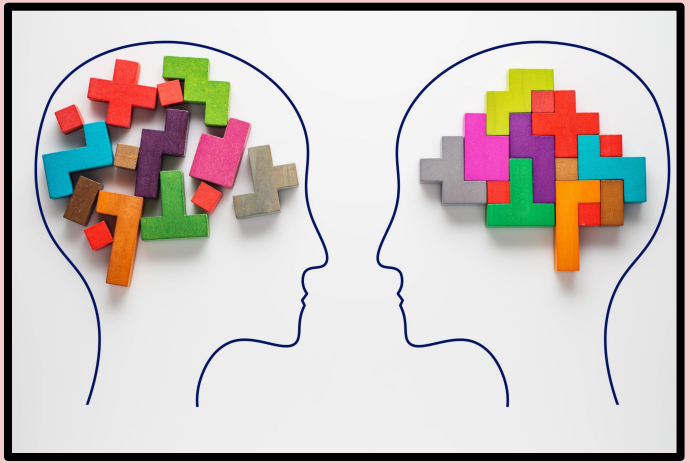
- **Purpose**
  - **create fully autonomous robot**
    - **solve any maze, without needing any help**

Introduction to the project.

- **Scope**
  - **Uses 1 Arduino Qwiic and 4 AA batteries.**
  - **Maze has left and right turns, varied straight sections**
  - **Only Arduino components (except rubber bands)**

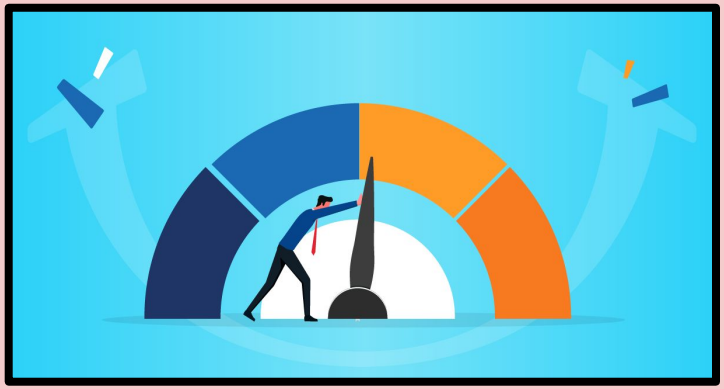
Objectives & Requirements

- **Functional Requirements**
  - **Green button = Start**
  - **Red button = Stop**
  - **Green LED = Robot moving**
  - **Red LED = Obstacle < 3 inches**



Objectives & Requirements

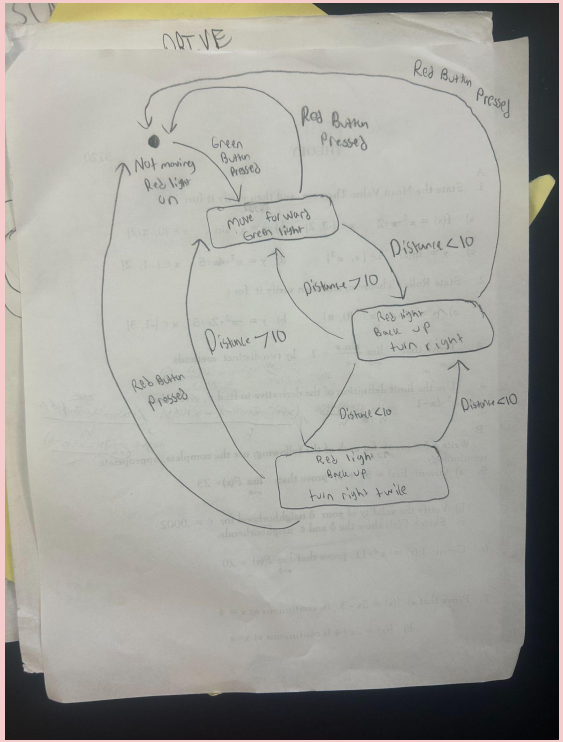
- **Performance Requirements:**
  - **Complete maze < 1 minute 30 seconds**
  - **No human intervention (-5 points)**
  - **Unable to solve maze is -15 points**
  - **Unable to move is -30**

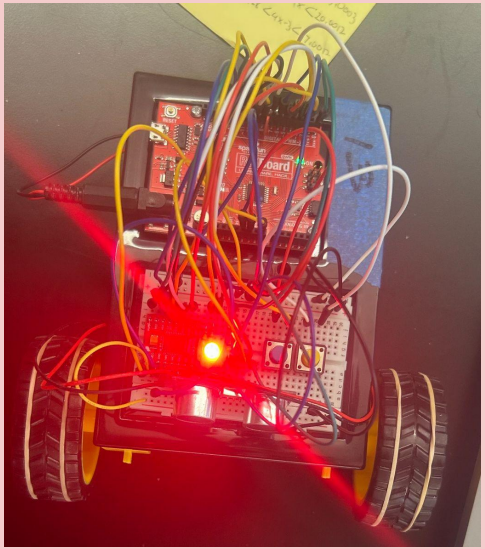


Objectives & Requirements

- **Constraints**
  - **Only Arduino components**
  - **Must be autonomous for an A grade**

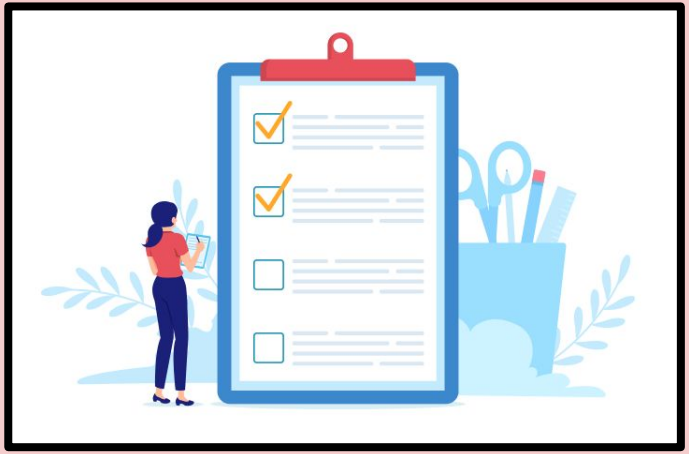
System Architecture and Design (Block Diagram)



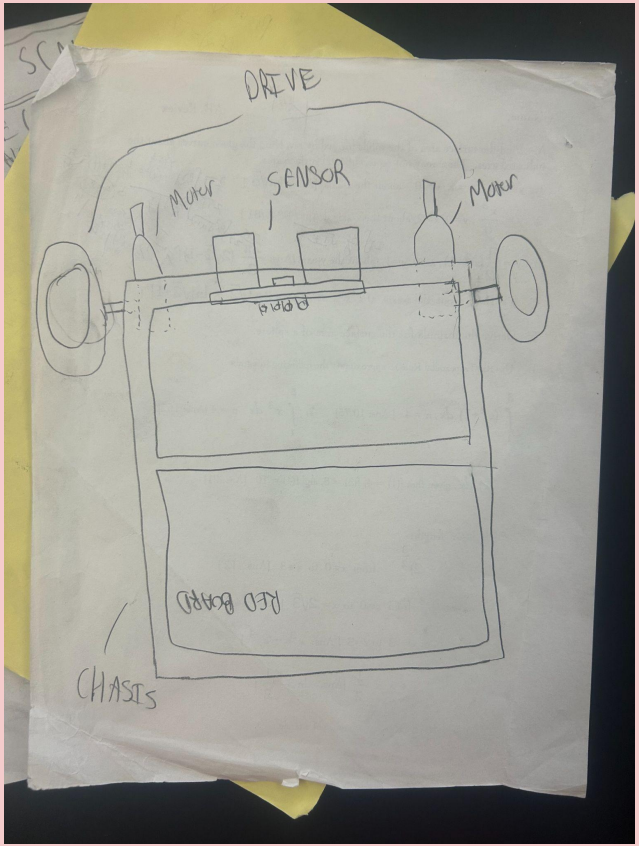


System Architecture and Design (Component List)

- **26 Jumper wires**
- **3 330Ω Resistors**
- **1 4 pin LED**
- **1 Distance Sensor**
- **2 Motors**
- **2 buttons**
- **1 Motor Driver**



System Architecture and Design (Chassis and Drive)



System Architecture and Design (Sensor Placement)

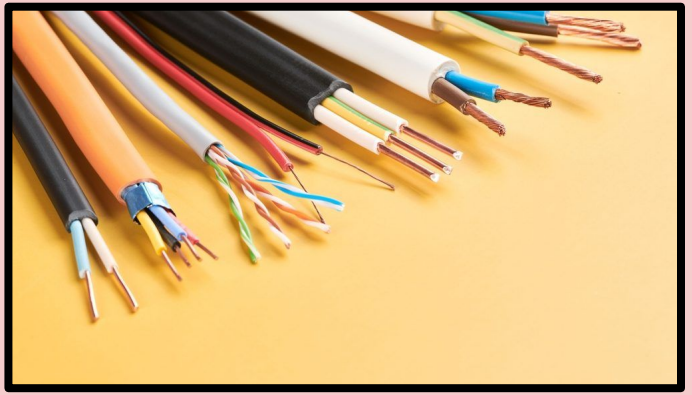
- **The Distance Sensor is placed with this orientation on these specific pins**
- **A14 - VCC**
- **A15 - Trig**
- **A16 - Echo**
- **A17 - GND**

System Architecture and Design (Electrical Wiring)

- **Pin Assignments:**
- **Jumper Wire (J7 - Digital Pin 10)**
- **Jumper Wire (E14 - 5V Rail +)**
- **Jumper Wire (E15 - Digital Pin 6)**
- **Jumper Wire (E16 - Digital Pin 5)**
- **Jumper Wire (E17 - GND Rail -)**
- **Jumper Wire (J18 - Digital Pin 4)**
- **Jumper Wire (E19 - GND Rail -)**
- **Jumper Wire (J20 - Digital Pin 3)**
- **Jumper Wire (J21 - Digital Pin 0)**
- **Jumper Wire (J22 - GND Rail -)**
- **Jumper Wire (J24 - Digital Pin 2)**

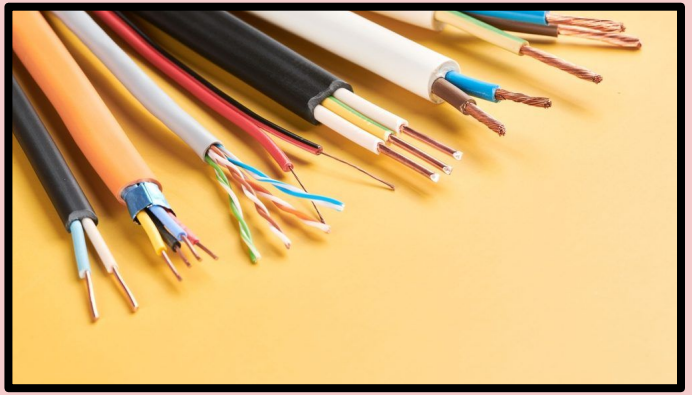
System Architecture and Design (Electrical Wiring)

- **Pin Assignments continued:**
- **Jumper Wire (J25 - GND Rail -)**
- **Jumper Wire (J27 - Digital Pin 7)**
- **Jumper Wire (5V Rail + - 5V (Red Board))**
- **Jumper Wire (GND Rail - -GND (Red Board))**
- **Jumper Wire (5V Rail + - 5V Rail +)**
- **Jumper Wire (5V Rail + - 5V Rail +)**
- **Motor Driver (C1 - C8) & (G1 - G8)**
- **Right Motor (Red - A4) & (Black - A5)**

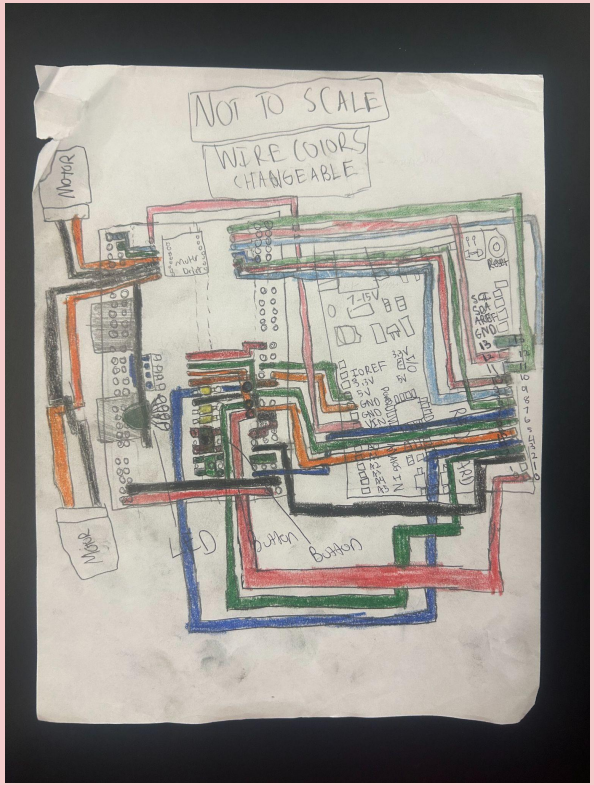


System Architecture and Design (Electrical Wiring)

- **Pin Assignments continued:**
- **Left Motor (Black - A6) & (Red - A7)**
- **Distance Sensor (A14(VCC), A15(Trig), A16(Echo), A17(GND))**
- **RGB LED (A18(Blue), A19(GND), A20(Green), A21(Red))**
- **330Ω Resistor (E18 - F18)**
- **330Ω Resistor (E20 - F20)**
- **330Ω Resistor (E21 - F21)**
- **Red Button (D22 - D24) & (G22 - G24)**
- **Green Button (D25 - D27) & (G25 -**

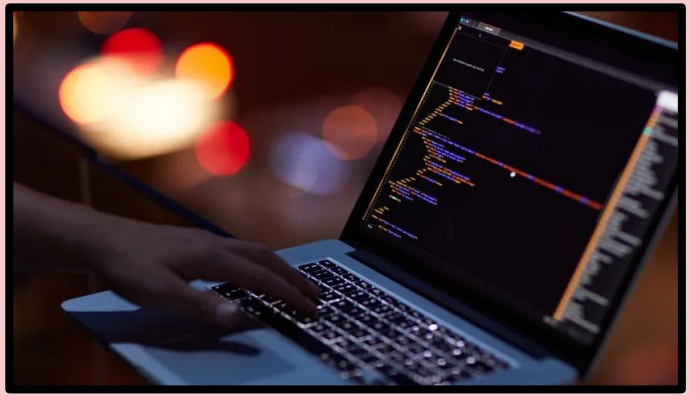


System Architecture and Design (Electrical Wiring, Diagram)

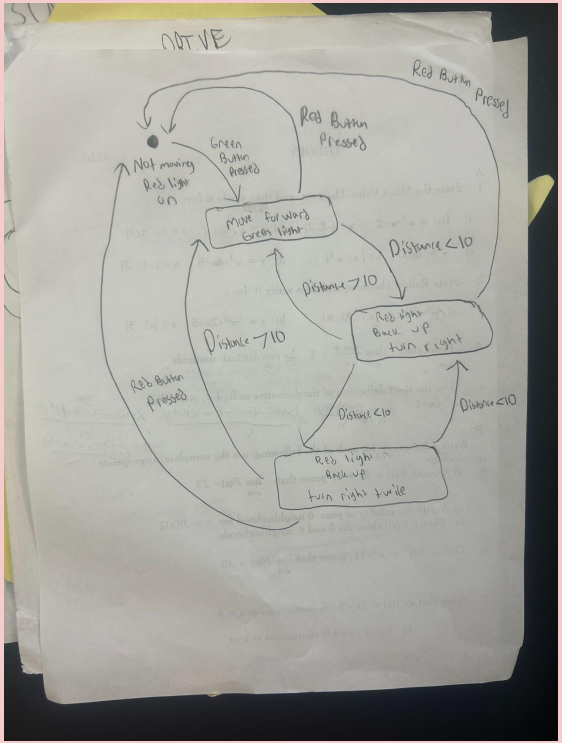


Maze-Solving Algorithm

- **Algorithm Description:**
  - **if no wall(green light on)**
    - **Move forward**
  - **If wall detected(red light on):**
    - **turn right**
    - **if wall**
      - **turn right again**
    - **Else**
      - **move forward**

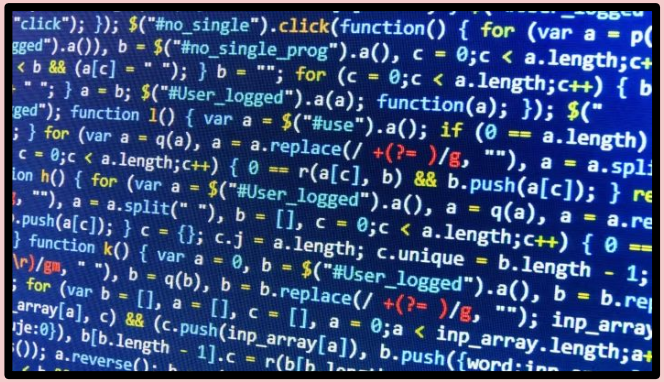


Maze-Solving Algorithm (Flowchart)



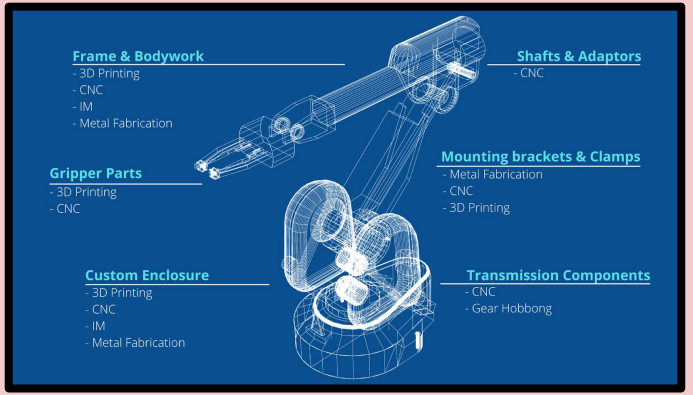
Maze-Solving Algorithm

- **Pseudocode**
  - if (start_button_pressed) {
  - if (distance < 3) {
  - Red_light();
  - turn_right();
  - if (distance < 3) {
  - Red_light();
  - turn_right_twice()
  - }
  - }
  - else {
  - Green_light();
  - move_forward();
  - }
  - }
  - else{
  - Stop moving();
  - Red_light();
  - }



Software Design

- **Module Breakdown:**
- **Sensor Reading - Detects object distance to guide robot actions.**
- **Motors - used to make the car move and turn**
- **Buttons - Start/stop robot based on input signals**

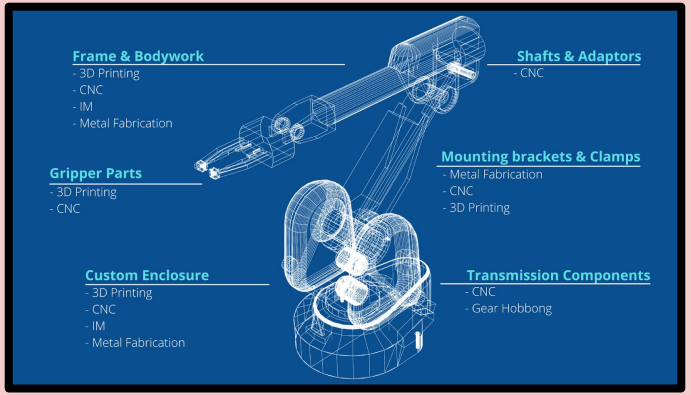


Software Design

- **Module Breakdown continued:**

- **LED - Changes color based on sensor and button input, green when moving, red when an object is close or stop is pressed.**

- **Motor Control - Sends commands to motors to control spin duration, direction, and speed.**



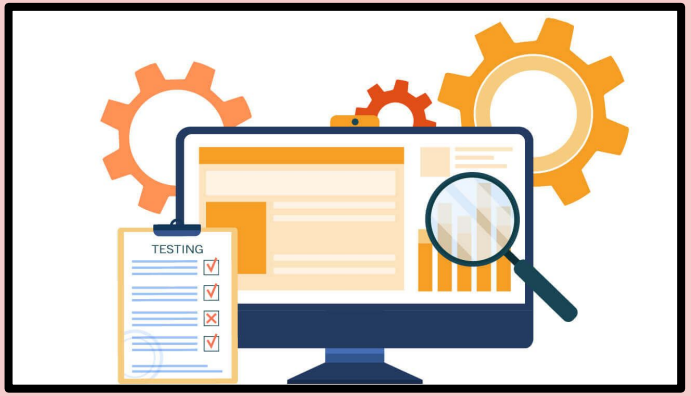
Testing & Validation

- **Test Plan:**

  - **Buttons: Press the on button to start movement and turn the LED green; press off to stop and turn the LED red.**

  - **LEDs: Verify initial red light, green when started, and red when an object is detected.**

  - **Motors: Place the car in the maze and confirm it makes all required turns.**



Testing & Validation

- **Success Criteria:**
- **On button starts the car (green LED, motors moving forward).**
- **Off button stops the car (red LED, motors off).**
- **If an object is detected, the car turns away and the LED turns red.**

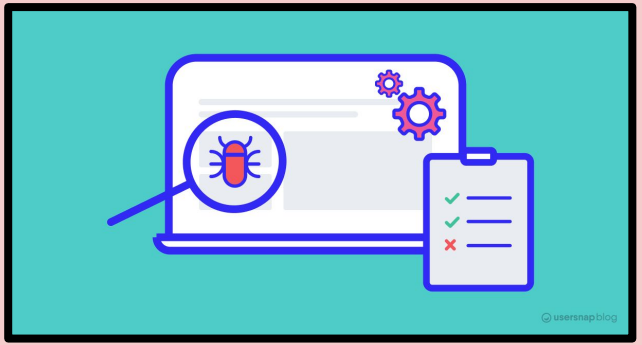
Testing & Validation

- **Results Log**

- **Test 1: LED does not work, Button Does not work, the car does work however and it can move in the maze, does hit walls and not turn effectively.**

- **Test 2: LED works, buttons dont work, the car does the same as Test 1**

- **Test 3: LED does not work, Buttons dont work, car does not move.**

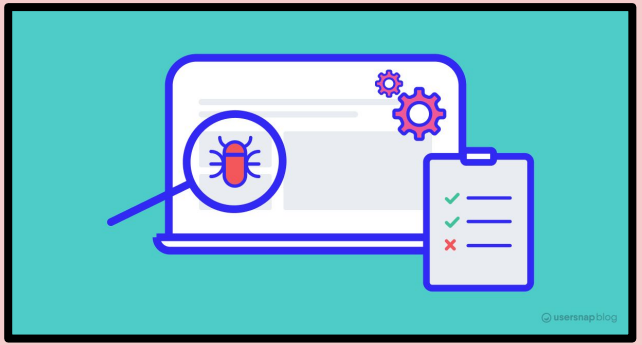


Testing & Validation

- **Results Log continued:**

- **Test 4: LED works, Buttons work, car is unable to turn.**

- **Test 5: LED works, Buttons work, car can move and turn. When placed in maze it starts going through the maze but gets stuck 25% of the way through**

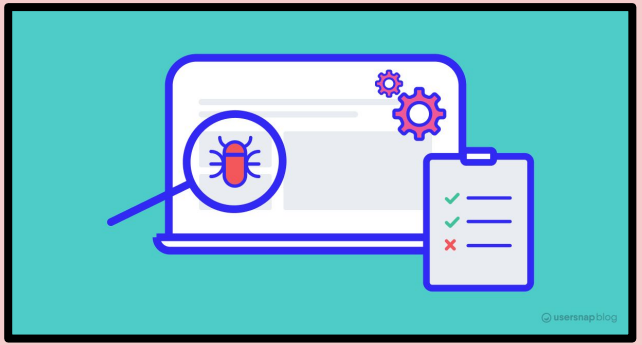


Testing & Validation

- **Results Log continued:**

- **Test 6: LED Works, Buttons work, car can move and turn. When placed in the maze it starts going through the maze and gets stuck 10% of the way through**

- **Test 7: LED Works, Buttons work, car can move and turn. When placed in the maze it starts going through the maze and is unable to make the 180 degree turn.**



Timeline & Milestones

- **5/9/25 - Research & design document**
- **5/16/25 - Basic navigation prototype**
- **5/23/25 - Full maze tests & tuning**

Summary of our experience

All in all I think that this experience was great, me and shahil worked a lot together on the robot and had lots of fun doing it.