



# **SUPER SONIC CAR**

BY : YONE L5

# YONE



**Attivich  
Saveevanlop  
(Yim)**



**Ratchapong  
Sertthikul  
(Dale)**

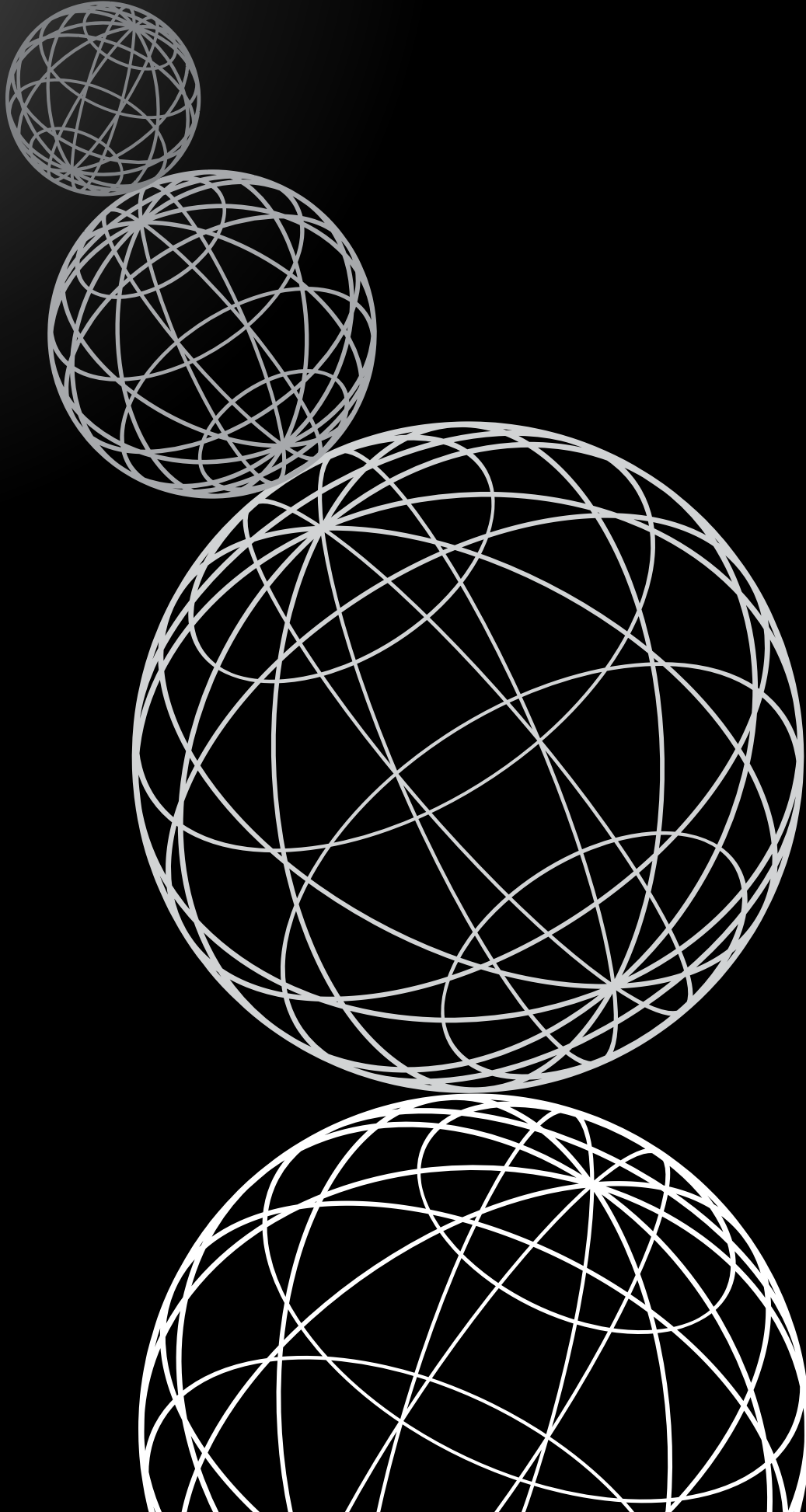


**Chaiwat  
Phumjaroen  
(Antony/Tong)**

# OVERVIEW

เป็นโปรเจกต์ที่ใช้รถในการตรวจสอบสภาพดินว่าพร้อมที่จะ  
เพาะปลูกหรือไม่โดยสามารถดูได้จากมือถือ และสามารถปรับ  
โหมดของรถให้เป็น auto และ manual ได้ โดยมี Ultrasonic  
สำหรับตรวจจับสิ่งกีดขวาง





# DETAIL/REQUIREMENT

- Arduino UNO + Motor Shield (L293D) = ควบคุมมอเตอร์ DC 2 ตัว
- ESP32 = WiFi AP, Web Server, WebSocket, Serial เชื่อม Arduino
- Ultrasonic Sensor (HC-SR04) = ตรวจจับสิ่งกีดขวาง (2–200 cm)
- Servo Motor (SG90) = หมุนสแกนเซนเซอร์ (0–180°)
- DC Motors (x4) = ขับเคลื่อนล้อซ้าย/ขวา
- Power Supply = 7.4–12V

# SPECIFICATION

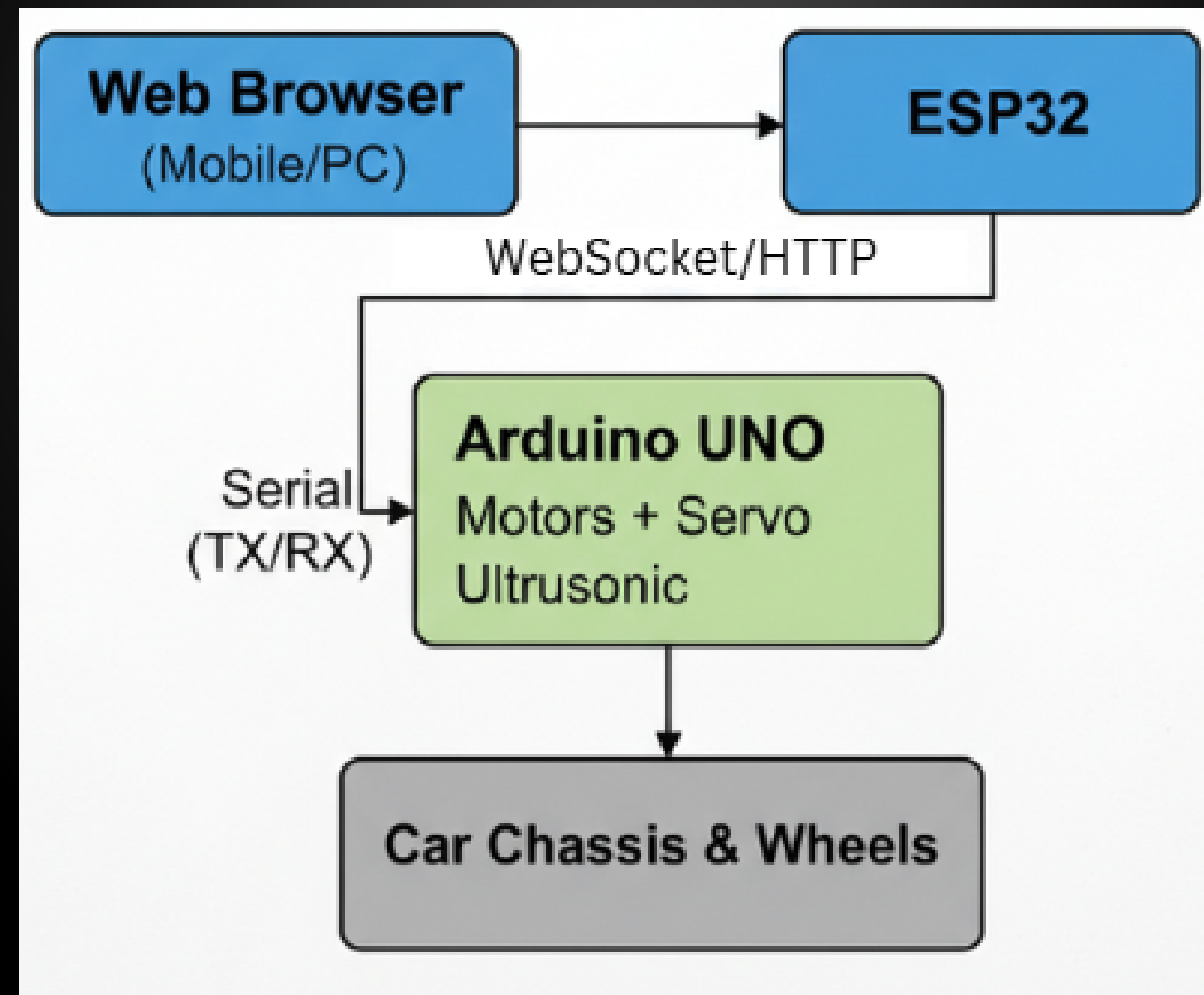
Response time < 200 ms

Battery life  $\geq$  30 นาที

โค้ดใช้ non-blocking (ไม่มี delay ยกเว้น  
setup)

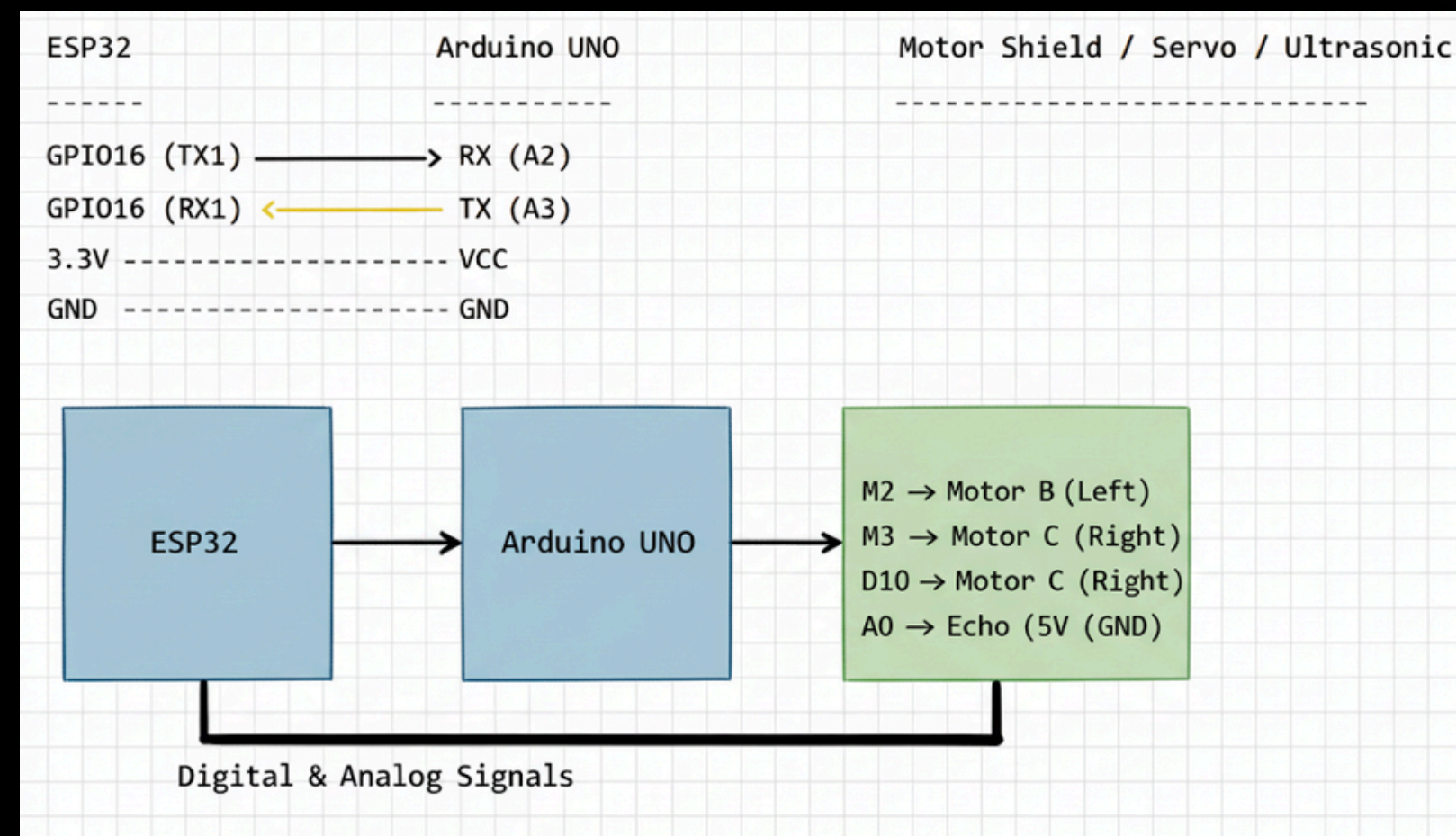
รองรับผู้ใช้หลายคน (WebSocket broadcast)

# ARCHITECTURAL DESIGN

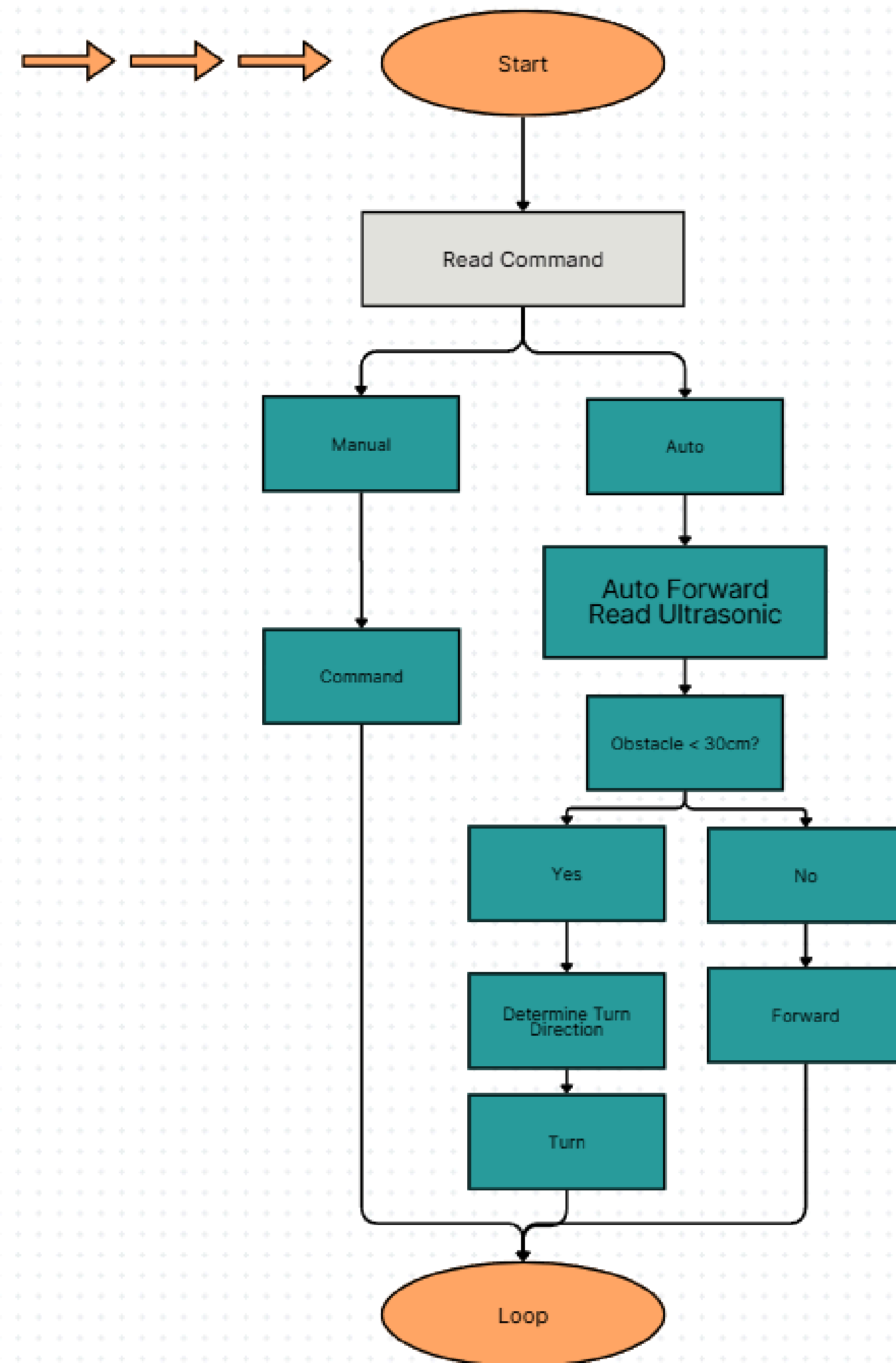




# DETAILED DESIGN



# FLOWCHART & CODE





## สร้าง AP และ WebSocket

```
const char* ssid="ESP32_Car";  
const char* password="12345678";  
WebServer server(80);  
WebSocketsServer websocket(81);
```

## หน้าเว็บควบคุม (HTML + JS)

- ปุ่ม **Forward / Backward / Left / Right**
- ปุ่ม **Manual / Auto**
- **JavaScript** = ส่งข้อความ **WebSocket**

## ฟังก์ชัน WebSocket Event

```
void websocketEvent(...) {  
  if(msg.startsWith("MODE:")) { Serial1.println(msg); }  
  else { Serial1.println(msg); }  
}
```

# ESP32 Code

## Setup และ Loop

```
void setup() {  
  WiFi.softAP(ssid,password);  
  server.on("/", handleRoot);  
  server.begin();  
  websocket.begin();  
  websocket.onEvent(websocketEvent);  
}  
void loop() {  
  server.handleClient();  
  websocket.loop();  
}
```

# Arduino UNO Code

## ตัวแปรโหมด

```
String mode = "MANUAL"; // โหมดเริ่มต้น
enum AutoState { AUTO_FORWARD, AUTO_SCAN, AUTO_TURN };
AutoState autoState = AUTO_FORWARD;
```

## การอ่าน Ultrasonic

```
int readPing() {
  int cm = sonar.ping_cm();
  if(cm == 0) cm = 250; // ถ้าไม่เจอสิ่งกีดขวาง
  return cm;
}
```

## รับคำสั่งจาก ESP32

```
if(ESPSerial.available()){
  String cmd = ESPSerial.readStringUntil('\n');
  if(cmd=="MODE:AUTO"){ mode="AUTO"; autoState=AUTO_FORWARD; }
  else if(cmd=="MODE:MANUAL"){ mode="MANUAL"; stopMotors(); }
  else if(mode=="MANUAL"){ handleManual(cmd); }
}
```

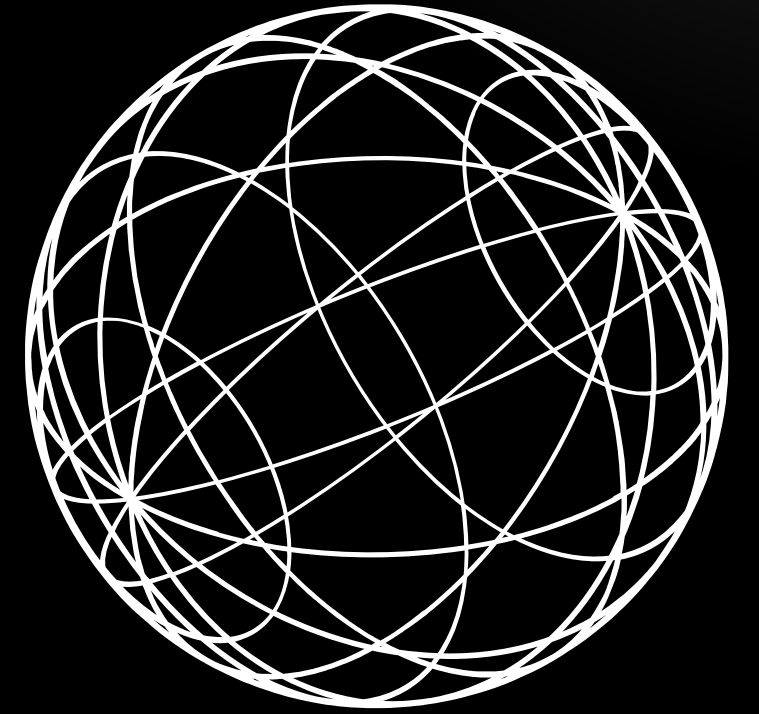
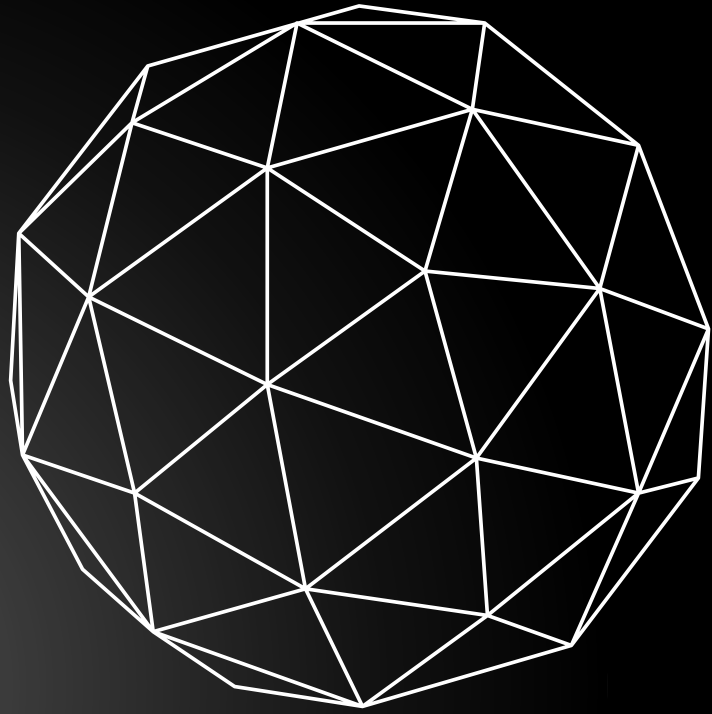
## ฟังก์ชันควบคุมมอเตอร์

```
void forward() { motorB.setSpeed(255); motorC.setSpeed(255);
motorB.run(FORWARD); motorC.run(FORWARD); }
void backward() { motorB.run(BACKWARD);
motorC.run(BACKWARD); }
void left() { motorB.run(BACKWARD); motorC.run(FORWARD); }
void right() { motorB.run(FORWARD); motorC.run(BACKWARD); }
void stopMotors() { motorB.run(RELEASE); motorC.run(RELEASE); }
```

## Auto Mode Logic

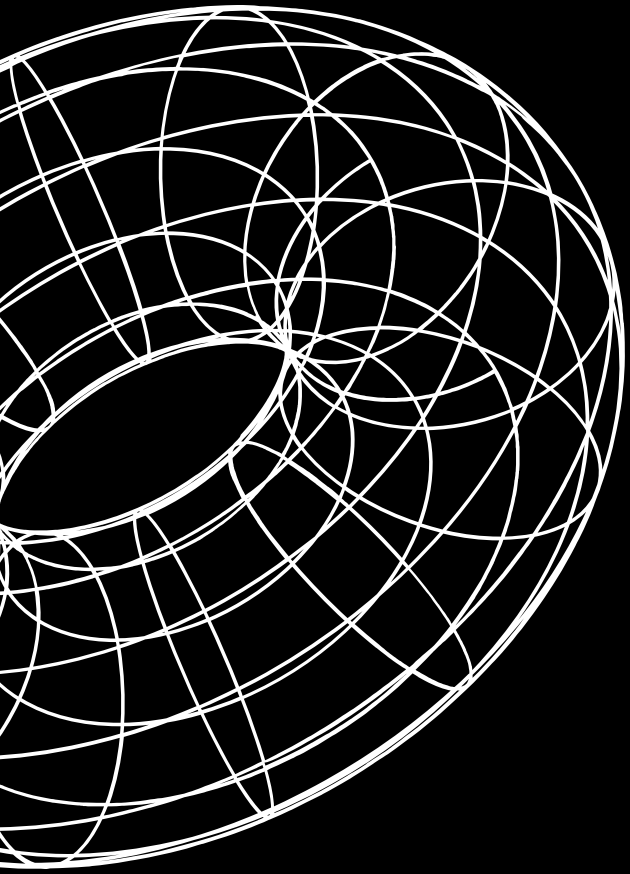
```
void autoDrive(unsigned long now) {
  switch(autoState) {
    case AUTO_FORWARD:
      forward();
      if(distance <= 30){ stopMotors(); autoState = AUTO_SCAN; }
      break;
    case AUTO_SCAN:
      if(scanDone){ startTurn(turnDir); autoState = AUTO_TURN; }
      break;
    case AUTO_TURN:
      if(turnDir==1) right();
      else if(turnDir==-1) left();
      else forward();
      if(now - stateStart >= 400){ stopMotors(); autoState = AUTO_FORWARD; }
      break;
  }
```

# PICTURE

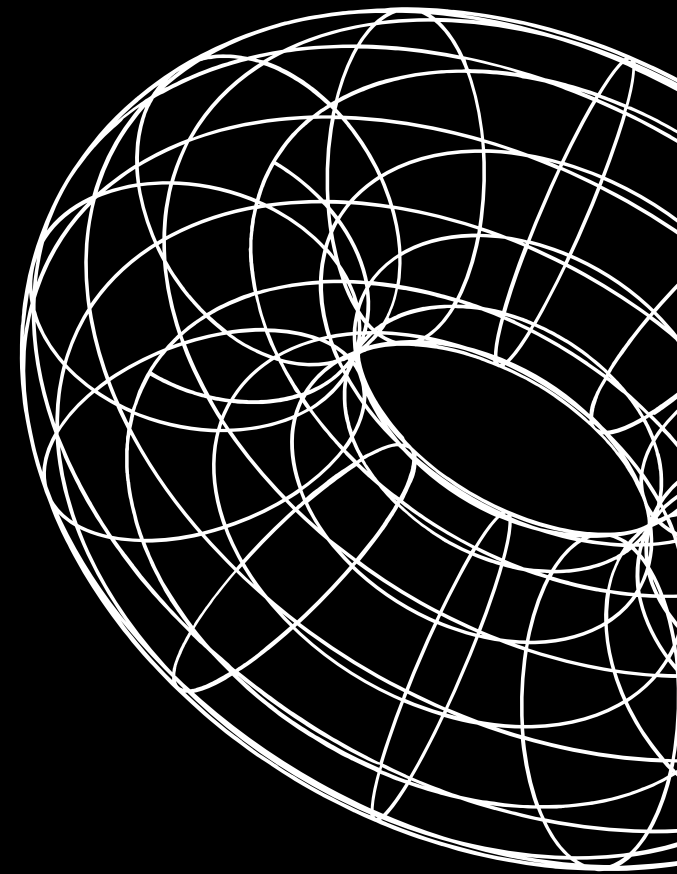




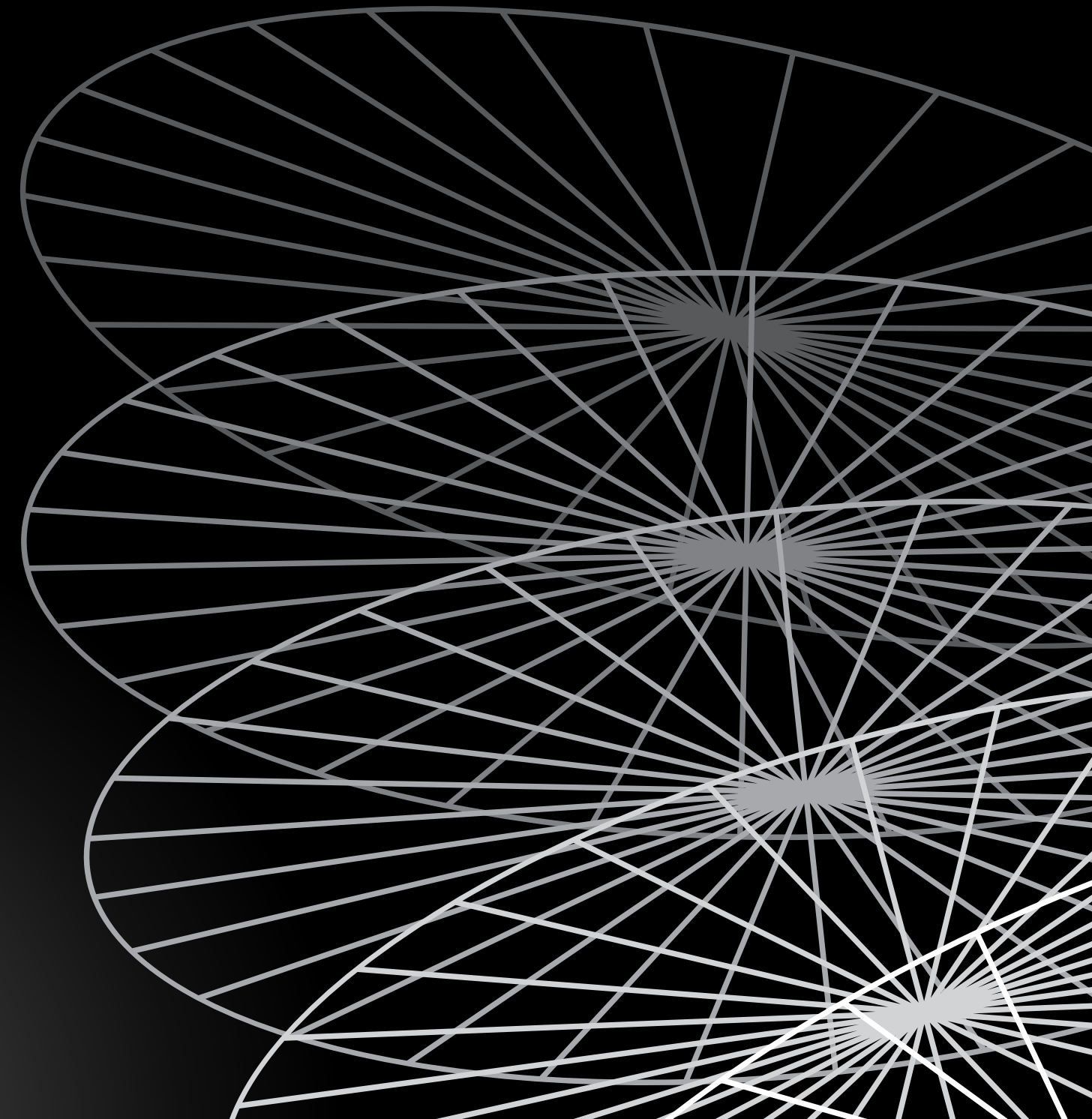
**VDO**



# **PROBLEM AND SOLUTION**



# GANTT CHART







**THANK YOU**

YONE L2