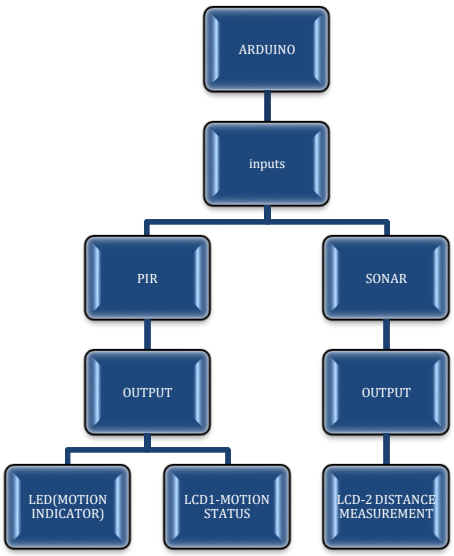
Project:	
Document:	Software Requirements Specification
Author:	Iddi Munkaila
Date:	21-11-24
Version:	1

Overall Description	
Background:	A Distance Motion Analyzer is a critical tool for measuring motion accurately over specified distances. By providing real-time data on an object's motion characteristics—position, speed, and acceleration—it becomes an essential device for applications in robotics, automotive testing, and dynamic system diagnostics. Its purpose is to enhance understanding, improve system design, and ensure safety by offering precise, reliable data for analyzing dynamic systems.
Project Overview:	A distance motion analyzer provides information on an object's position, speed, and motion patterns by tracking and measuring its movement across a predetermined range.
Block Diagram:	

Block Descriptions:

PIR Sensor: Detects motion and sends signals to the Arduino.
Ultrasonic Sensor: Measures object distance and sends data to the Arduino.
Arduino Board: Processes input from sensors and controls outputs.
LCD 1: Displays motion status (e.g., "Motion Detected").
LCD 2: Displays distance measurements (e.g., "Object at X cm").
LED: Lights up to indicate motion detected

Typical User:

Characteristics:

- **Experience**: Basic technical knowledge, familiar with sensors and Arduino.
- **Purpose**: Home automation, security, or education.
- **Demographics**: 15-40 years old, tech enthusiasts, or students.
- **Behavior**: Prefers real-time feedback, user-friendly interface, and clear error handling.
- **Environment**: Indoor settings like homes, labs, or classrooms.

Constraints:

Hardware:

- Limited sensor range and accuracy.
- Potential hardware malfunctions (e.g., loose wires, damaged components).
- Power supply issues affecting stability.

Software:

- Arduino's limited memory and processing capacity.
- Handling real-time inputs may cause delays.

Environmental:

- PIR sensors are affected by temperature and interference.
- Ultrasonic sensors are impacted by reflective or sound-absorbing surfaces.

User:

- Requires basic technical knowledge for setup and troubleshooting.

Cost:

- Budget limits may affect component quality.

Specific Requirements

External Interfaces:

Inputs:

1. **Sensors**:
   - **PIR Sensor**: Detects motion and sends HIGH/LOW signal.
   - **Ultrasonic Sensor**: Measures distance to nearby objects.
2. **User Inputs**:
   - Physical interaction for resetting, calibrating, or powering the system.
3. **Data Files (Optional)**:
   - Predefined thresholds or settings loaded into the Arduino from a connected device.

Outputs:

1. **Devices**:
   - **LCD 1**: Displays motion status (e.g., "Motion Detected").
   - **LCD 2**: Displays distance information (e.g., "Object at 30 cm").
   - **LED**: Visual indicator for motion detection.
2. **User Feedback**:
   - Outputs on LCDs and LED are directly observable by the user.
3. **Data Files (Optional)**:
   - Logged sensor data sent to a computer via serial communication for further analysis.

Specific Requirements: Below are the MuSCoW requirements

MuSCoW

Number	Requirement	MuSCoW

1	Platform requirements: Arduino microcontroller powered by a PC or laptop.	Must
2	Interface requirements: LCD displays and LEDs for user output.	Must
3	Sensor input range: PIR (5-7 meters), Ultrasonic (2-400 cm).	Must
4	Calculation accuracy: ±2 cm for distance measurements.	Must
5	Visual output: Clear formatting on LCDs with refresh rates under 500ms.	Should
6	Data storage: Temporary storage of readings in memory for display.	Should
7	System response time: Motion and distance detection within 1 second.	Must
8	Optional: Bluetooth or Wi-Fi communication for remote data viewing.	Could
9	Cloud storage for logs of motion/distance data.	Won't