Hamdard University

Department of Computing

Final Year Project

Luma Bot: AI Integrated Robot

Submitted by
Rayyan Shaikh 2277-2021
M.Asad 1613-2021
Syed Ibtesam Ahmed 2174-2021

Supervisor(s)
Dr. Khurram
Iqbal

Fall 2021

Document Sign off Sheet

Document Information

Project Title	Luma Bot:AI Integrated Robot
Project Code	FL-023/24
Document Name	Software Design Specifications
Document Version	1.0
Document Identifier	SDS
Document Status	Final
Author(s)	Syed Ibtesam Ahmed
Approver(s)	Dr. Khurram Iqbal
Issue Date	11-1-2025

Name	Role	Signature	Date
Rayyan Shaikh	Team Lead		
M.Asad	Team Member 2		
Syed Ibtesam Ahmed	Team Member 3		
Dr. Khurram Iqbal	Supervisor		

Revision History

Date	Version	Description	Author
11-1-2025	1.0	Overview of project	Syed Ibtesam Ahmed

Definition of Terms, Acronyms, and Abbreviations

AI (Artificial Intelligence): The simulation of human intelligence processes by machines, especially computer systems, involving learning, reasoning, and self-correction.
Computer Vision: A field of AI that enables computers to interpret and make decisions based on visual data from the world.
IoT (Internet of Things): A network of physical devices embedded with sensors, software, and other technologies to connect and exchange data with other devices and systems over the internet.
ML (Machine Learning): A branch of AI focused on building systems that can learn from and make decisions based on data.
Ardiuno : A small, affordable computer used for programming and electronic projects, including the development of smart devices.
UC (Use Case): A list of actions or event steps typically defining the interactions between a role (actor) and a system to achieve a goal.
User: In the context of this project, a visually impaired individual using the Smart Vision Kit.
Visually Impaired Individual: A person with significant visual impairment that cannot be corrected fully with glasses or contact lenses, affecting their ability to perform daily tasks.
Audio Feedback: Auditory information provided to the user describing detected objects and their surroundings.
Real-Time Processing: The immediate processing of data as it comes in, without any significant delay, to provide timely feedback or results

Table of Contents

1 Introduction
1.1 Purpose of Document
1.2 Intended Audience
1.3 Project Overview
1.4 Scope
2 Design Considerations
2.1 Assumptions and Dependencies
2.2 Risks and Volatile Areas
3 System Architecture
3.1 System Level Architecture
3.2 Software Architecture
4 Design Strategy
5 Detailed System Design
5.1 Database Design
5.1.1 ER Diagram
5.1.2 Data Dictionary
5.1.2.1 Data 1
5.1.2.2 Data 2
5.1.2.3 Data n
5.2 Application Design
5.2.1 Sequence Diagram
5.2.1.1 <Sequence Diagram 1>
5.2.1.2 <Sequence Diagram 2>
5.2.1.3 <Sequence Diagram n>
5.2.2 State Diagram
5.2.2.1 <State Diagram 1>
5.2.2.2 <State Diagram 2>
5.2.2.n <State Diagram n>
6 References

1. Introduction

1.1 Purpose of Document

This document outlines the technical design specifications of the LumaBot project. It defines the system architecture, software components, control logic, and interaction between modules used to create an AI-powered cleaning robot.

1.2 Intended Audience

- Project Supervisor and Coordinator
- Final Year Evaluation Committee
- Development and Testing Team

1.3 Project Overview

LumaBot is an AI-integrated autonomous robot designed to detect and clean small debris such as dust, wrappers, and cigarette butts. It uses an ESP32-CAM for basic visual input and communicates with a second ESP32 board to control the motors, vacuum system, and obstacle detection using ultrasonic sensors. The robot operates independently, navigating indoor spaces and performing cleaning tasks without the need for external control.

1.4 Scope

The document focuses on embedded software architecture, control strategies, hardware integration, and testing plans.

---

2. Design Considerations

2.1 Assumptions and Dependencies

- Adequate indoor lighting is available for object detection.
- The robot operates on a flat and obstacle-contained surface.

2.2 Risks and Volatile Areas

- False negatives during debris detection may cause cleaning to fail.
- Hardware limitations could affect real-time processing.
- Ultrasonic sensors may misread reflective or curved surfaces.
- Continuous operation may lead to power depletion or overheating.

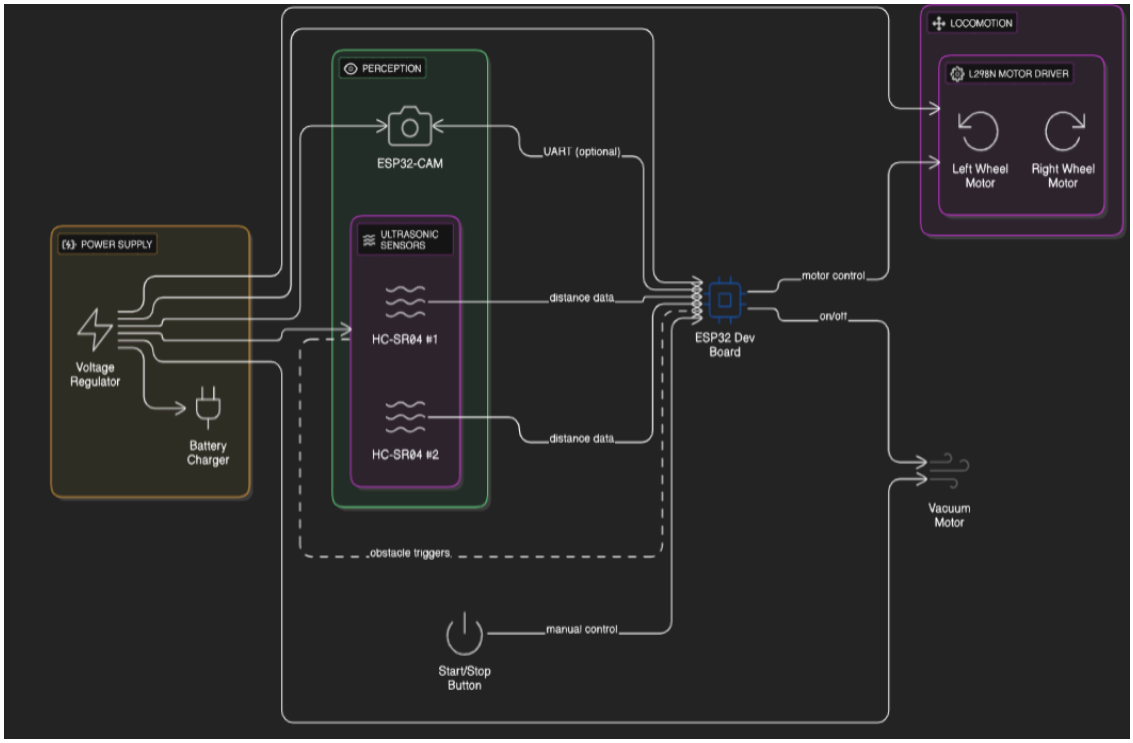
---

3. System Architecture

3.1 System Level Architecture

The system uses a dual-microcontroller setup:

- ESP32 Dev Board manages vacuum activation via relay, movement control via motor driver, and obstacle avoidance via ultrasonic sensors.



3.2 Software Architecture

The software is divided into modular components:

- Arduino-based control code for movement, vacuum, and Bluetooth.
- Serial or GPIO-based signaling between the ESP32 boards.

---

4. Design Strategy

- **Modularity:** Software and hardware components are isolated for independent development and debugging.
- **Safety:** Sensors trigger real-time responses to avoid crashes.
- **Control:** Vacuum is continously activated and suck debris.

---

5. Detailed System Design

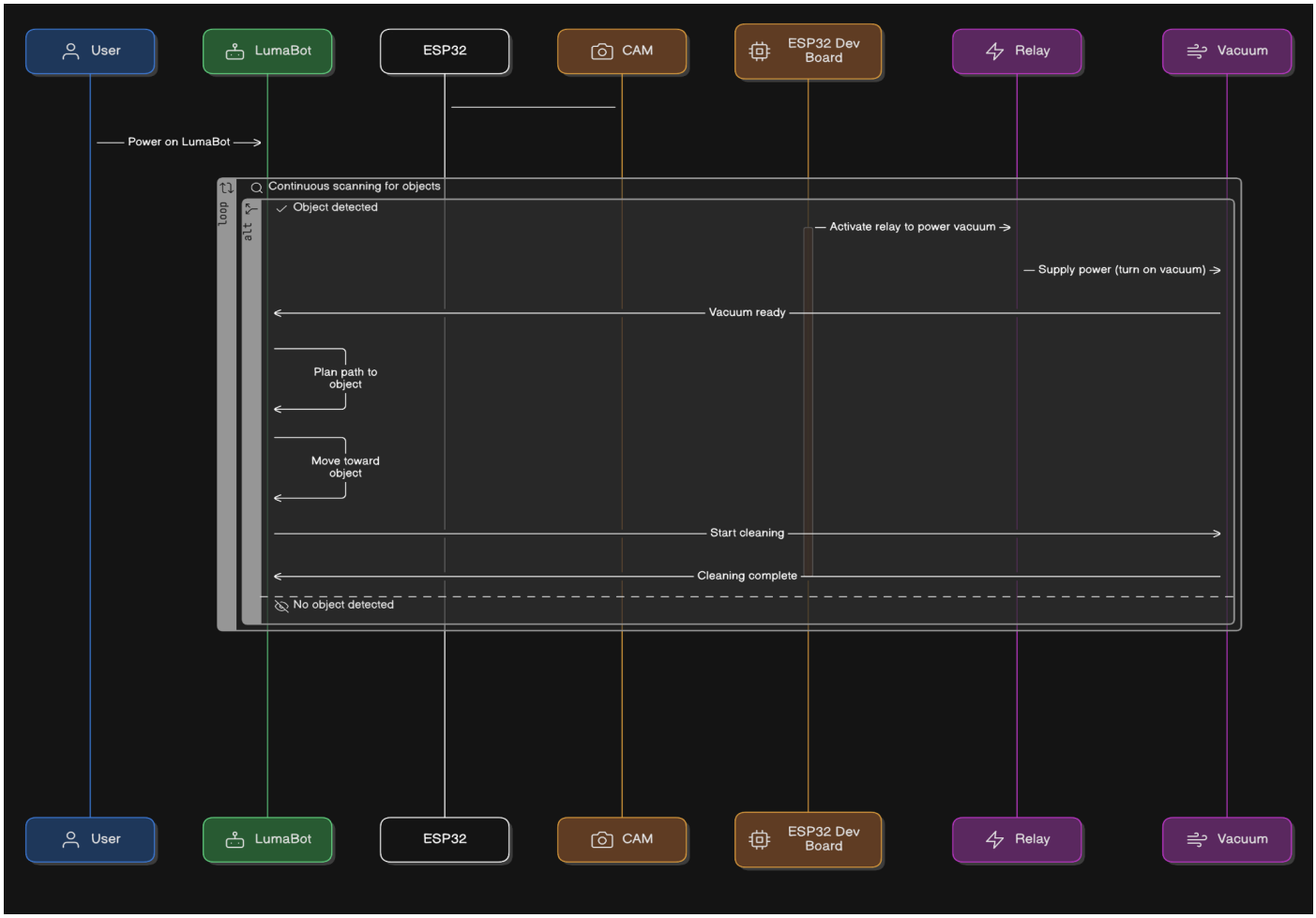
5.1 Database Design

LumaBot operates as a real-time embedded system and does not use a traditional database. Any future enhancements for data logging can include local SD card storage or cloud integration.

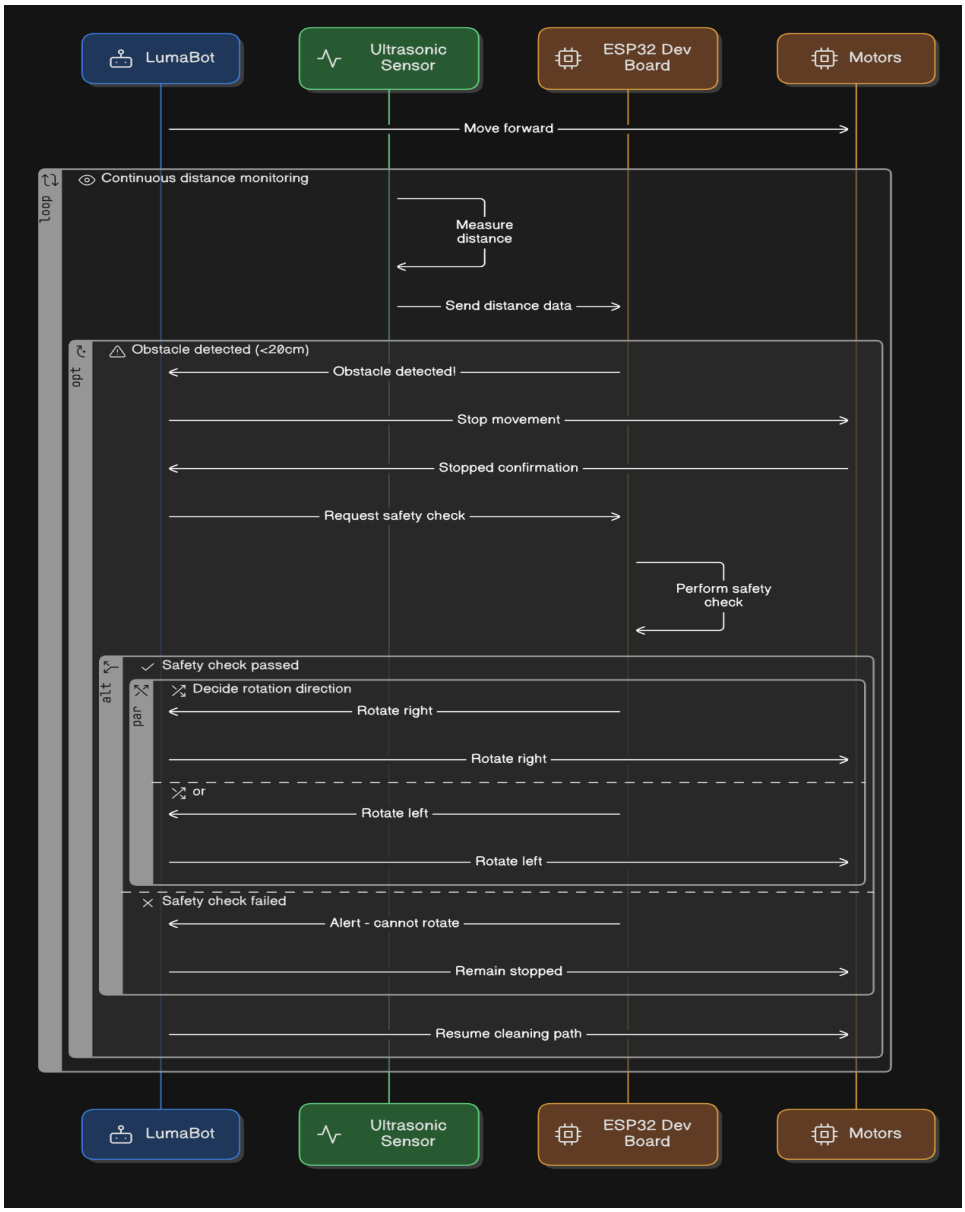
5.2 Application Design

5.2.1 Sequence Diagrams

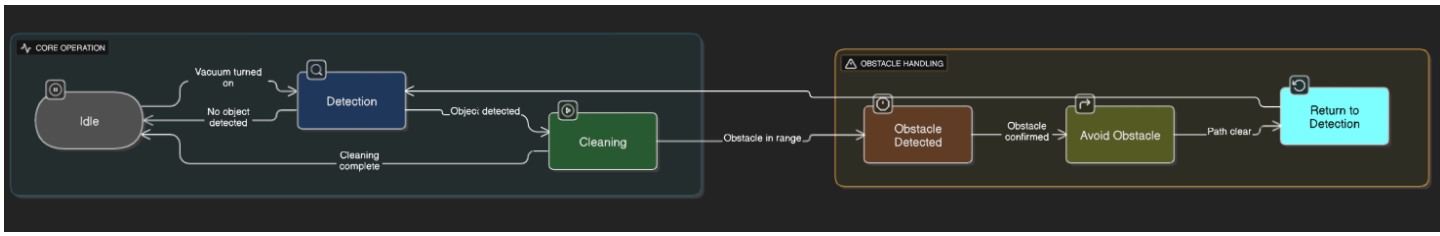
5.2.1.1 Detection and Cleaning Flow



5.2.1.2 Obstacle Avoidance Flow



5.2.2 State Diagrams



---

6. References

1. Arduino. (n.d.). *ESP32 Board Guide*. https://docs.arduino.cc/hardware/esp32
2. Espressif Systems. (n.d.). *ESP32-CAM Documentation*. https://www.espressif.com/
3. Ultralytics. (2020). *YOLOv5 GitHub Repository*. https://github.com/ultralytics/yolov5
4. Circuit Digest. *ESP32 Relay Interface*. https://circuitdigest.com/
5. LabelImg. *Image Annotation Tool*. https://github.com/tzutalin/labelImg
6. TutorialsPoint. *Agile Software Development*. https://www.tutorialspoint.com/agile/
7. W3Schools. *Bluetooth with Arduino*. https://www.w3schools.com/arduino/