

Final Project Proposal: Embedded Systems

Deadline: Tuesday, Feb 10

Format: PDF or Printed Copy

Instructor Approval: Required before starting.

1. Team Information

Group Name: Imprompter

Member Name	Role (Hardware / Backend / Frontend / AI)
1. Encomienda, Francis Frederick	Hardware, Backend
2. Valdez, Jasper Lou	AI, Frontend

2. Project Overview

Project Title: Lumos: The AI Environment Companion

The Elevator Pitch:

Lumos is a portable, AI-driven workspace companion powered by the Arduino R4 WiFi. Unlike static sensor kits, Lumos adapts to its environment. Through a 'Chat with Room' interface, the user can tell the system its new location (e.g.,), allowing Google Gemini AI to recalibrate its analysis of noise, light, and temperature to suit that specific context

Problem Statement:

Standard IoT devices use fixed thresholds (e.g., 'If Light < 300, turn on LED'). This fails when the device moves; 300 lumens is 'bright' for a bedroom but 'dim' for a library. We need Generative AI to understand these changing contexts. By chatting with the device to update its location, the system can dynamically shift its 'comfort' definitions without requiring code changes.

3. Technical Stack

A. Hardware (The Embedded Side)

Microcontroller: Arduino Uno R4 WiFi

Inputs (Sensors):

- Photoresistor (Light Level)
- Thermistor (Temperature)
- Push Button (Manual "AI Query" Trigger)

Outputs (Actuators):

- Built-in 12x8 LED Matrix (Unique R4 feature for scrolling text)
- RGB LED (Visual Mood Indicator)
- Active Buzzer (Audible Alerts for extreme conditions)

Power Source: USB-C (via Laptop/PC)

B. Software (The Web Side)

- Frontend: Tailwind CSS / HTML / JavaScript
- Backend: PHP
- Database: MySQL
- Hosting: Localhost (XAMPP/WAMP)

C. Artificial Intelligence (The Brain)

- AI Model: (e.g., Google Gemini API, OpenAI GPT-4o-mini)
- AI Task: (Check one)
 - [] Generative: Converting text to hardware commands (Natural Language Processing).
 - [] Classification: Deciding if data is "Good" or "Bad" (Sentiment Analysis).
 - [] Summarization: Reading logs and explaining what happened.
 - [] Expert System: Providing specific advice based on sensor data.
- Sample Prompt:
(Write the actual prompt you will send to the AI. Example: "Analyze this text: '\$user_input'. Return a JSON object with 'color_hex' and 'motor_speed' based on the emotion.")

4. System Logic (The Flow)

Scenario A: Sensor Monitoring

- Input: Sensor reads "Temp: 30°C, Light: 20% (Dim)".

- **Processing:** PHP sends this data to Google **Gemini API** via a custom prompt.
- **AI Decision:** The AI replies with: **JSON { "color": "#FF0000", "message": "Too Hot! Break Time." }.**
- **Actuation:** Arduino reads the database via HTTP GET and turns on the **LED Matrix to scroll "Too Hot!" and sets RGB LED to Red.**

Scenario B: Chat & Location Update

- **Input:** User moves the device and types "**I am now in the bedroom**" into the Chat tab.
- **Processing:** PHP stores "bedroom" as the active current_context in the database.
- **AI Decision:** Gemini acknowledges the shift: *"Understood. Switching to 'Relaxation Mode' thresholds. I will warn you if the lights are too cool/blue."*
- **Actuation:** The Arduino R4 flashes **BLUE** twice to confirm the mode switch.

5. Bill of Materials (Budget)

Component	Quantity	Status (Have it / Need to buy)
Arduino R4 WiFi	1	Have
Breadboard	1	Have
Photoresistor	1	Have
Thermistor	1	Have
RGB LED	1	Have
Active Buzzer	1	Have
10k Resistors	1	Have
Push Button	1	Have
Jumper Wires	1	Have

6. Timeline (Milestones)

- **Feb 13:** Proposal Submission.

- **Feb 14:** Hardware Circuit Building & Testing (Wiring).
- **Feb 14:** Web Dashboard & Database Setup.
- **Feb 15:** AI API Integration (PHP script working).
- **Feb 20 :** Full System Integration (Arduino talking to Database).
- **Finals Week:** Final Presentation & Demo.

7. Instructor Feedback (Leave Blank)

- **Approved:** [] Yes / [] No

- **Notes:** _____