

Using Ai for feedback and Improvement

❖ Refining logic

- **Prompt:** What can be done to improve any potential issue with the implementation of the logic system in a real-world setting?
- **Response:**
 - If the power goes out or something breaks, the gates should automatically fall to block cars. Safety first.
 - **Use Multiple Sensors:** Use different types (radar, cameras, in-ground loops) to check for trains and cars. If two out of three agree, then act. This prevents one broken sensor from causing a mistake.
 - **Big Battery (UPS):** Install a large battery backup to keep the system running during a short power cut.
 - **Self-Check & Alert:** The system should constantly check itself. If it finds a problem, it must immediately send an alert to the maintenance team and flash warning lights.
 - **In short:** Assume things will break. Add backups for everything. Always default to the safest option (gates down).
- **Outcome:**
 - With the help of the Ai's insight, use of multiple sensor and backup battery usage for system logs and operating system helped reduce potential risk or system error which could have led to casualty increasing safety and increasing durability in different environmental situations.
- **Influence**
 - The insight provided by the artificial agent led to implementation of backup battery and failsafe overriding option which increased safety and allowed the system to be able to operational in remote area where electricity is not available constantly.

❖ **Creating alternative solutions**

➤ **Prompt: What are some other possible solutions?**

➤ **Response:**

- **Build an overpass/underpass – Remove the crossing completely so cars and trains never meet.**
- **Communication with trains – Train sends its location/speed to the crossing to predict arrival.**
- **Remote monitoring – CCTV for operators to check and respond if something is stuck**

➤ **Outcome:**

- **Communication with the train increases safety and reduces the possibility of an accident in case there is a system failure which creates false response**

➤ **Influence:**

- **Adding the communication feature in the logic system as a confirmation step with system override feature for any manual intervention required.**

❖ Improving documentation

➤ **Prompt:** Help me create a Readme.md file for the project.

➤ **Response:**

- **Railway Crossing Gate Control System**
- **A robust, fail-safe logic controller designed to automate railway crossing gates, ensuring maximum safety through redundant sensors and fault-tolerant design.**

▪ **## 🛡️ Core Logic Principle**

- ****Gates lower when it's unsafe:****
- **- Train is approaching **OR****
- **- Vehicle is detected on the tracks**
- ****Gates raise only when it's safe:****
- **- No train is approaching **AND****
- **- No vehicle is on the tracks **AND****
- **- System confirms all sensors are functional**

▪ **## 🌐 Real-World Implementation Features**

- **- **Fail-Safe Design:** Gates default to ****closed**** during power failures**
- **- **Sensor Fusion:** Combines radar, LiDAR, and inductive loops for reliability**
- **- **2-out-of-3 Voting:** Requires multiple sensor agreement before action**
- **- **Battery Backup:** 72-hour UPS for power outage protection**
- **- **Self-Diagnostics:** Continuous health monitoring and alerting**

➤ **Outcome:** ChatGPT created a code format for the readme.md which translated the data I created into code which could be copied and pasted into the GitHub repository

➤ **Influence:** Organized data into a simple format which could be used to create the readme.md file in GitHub.

