Proposed Solution

1. System Overview

- **Objective**: Detect incoming trains using camera footage and alert workers to ensure safety.
- **Key Components**:
  - **Camera**: High-definition cameras installed at key points along the track.
  - **Machine Learning Model**: A train detection model trained to identify trains in the footage.
  - **Alert System**: Audio and visual alarms triggered upon detection.

2. Machine Learning Pipeline

- **Data Collection**:
  - Collect video and images of trains from different angles, distances, and lighting/weather conditions.
  - Include non-train scenarios to reduce false positives.
- **Preprocessing**:
  - Annotate the data (bounding boxes for trains).
  - Enhance data using augmentation techniques to simulate various conditions like rain, fog, or night.
- **Model Selection**:
  - Use an object detection model like YOLO (You Only Look Once) or SSD (Single Shot MultiBox Detector).
  - Train the model with annotated data to detect trains with high accuracy.
- **Testing and Validation**:
  - Evaluate the model on unseen data to ensure it generalizes well.
  - Optimize for low latency to detect trains in real-time.

3. Hardware Setup

- **Camera Selection**:
    - Weatherproof cameras with night vision capabilities.
    - High FPS to capture fast-moving trains.
- **Processing Unit**:
    - Use edge devices (e.g., NVIDIA Jetson Nano) for real-time inference.

4. Alert System

- **Audio Alerts**:
    - Install speakers near the track to broadcast warnings.
- **Visual Alerts**:
    - Install flashing lights for additional warnings.

5. Additional Features

- **Obstacle Detection**: Train the model to detect objects on the track (e.g., tools, equipment).
- **Monitoring Dashboard**: Provide a dashboard to monitor train detections and alerts in real-time.

6. Implementation Steps

1. **Prototype Development**:
    - Develop the detection model using a subset of the data.
    - Test it with a sample camera setup.
2. **Integration**:
    - Integrate the detection system with the alert mechanism.
3. **Deployment**:
    - Deploy cameras and edge devices at track repair sites.
    - Perform field testing to refine the system.
4. **Maintenance**:
    - Regularly update the model with new data for improved accuracy.
    - Inspect and maintain cameras to ensure functionality.

7. Challenges and Mitigation

- **Weather Conditions**:
  - Use robust data augmentation during training to handle weather variations.
- **Lighting Issues**:
  - Use cameras with infrared capabilities for low-light scenarios.

8. Benefits

- Eliminates water damage issues faced by sensors.
- Provides real-time detection and alerts.
- Enhances the safety of workers and trains.

9. Budget Estimate

Hardware Requirements and Costs

1. **CCTV Camera with IR**: Rs. 2000-5000 per unit.
2. **LoRa RF Module**: Rs. 1000.
3. **NVIDIA Jetson Nano 4GB Module**: Rs. 23,999
   - Link to Product
4. **SD Card 32GB and SD Card Reader**: Rs. 900.
5. **Power Supply Adaptor for NVIDIA Jetson**: Rs. 883
   - Link to Product
6. **Hardware Accessories**: Rs. 2000 (**breadboard, jumping wires, and other components for prototyping**).

10. Offline Alert Sending (Without Internet)

- Utilize the NVIDIA Jetson Nano's processing capability to run the model locally without requiring internet connectivity. This ensures uninterrupted alerts.
