Challenge Description: Smart Shelf Prototype Development

Objective: Develop a prototype of a smart shelf system that uses machine learning and integrated system architecture. The system should differentiate between retail products that are visually similar but vary slightly in size, and manage a virtual cart that updates in real-time based on product interactions.

Specific Task Requirements:

1. **Machine Learning Model Development**:
    - **Goal**: Develop a model that can accurately identify and differentiate between three sizes of the same product design. For example, Pepsi cans that come in 330ml, 250ml, and 150ml. The size difference might be minimal and should be detectable even if it is just a few millimeters apart.
    - **Technology**: Utilize computer vision technologies such as OpenCV for video processing and TensorFlow or PyTorch for machine learning.
    - **Dataset**: You may use your camera to create a dataset of the products being moved to and from the shelf, capturing various angles and lighting conditions.
2. **System Architecture**:
    - **Backend**: Implement backend logic to process data from the camera feed, analyze it through the machine learning model, and update the virtual cart status.
    - **Frontend**: Develop a simple yet effective HTML/CSS/JavaScript interface that displays the virtual cart. It should update in real-time as items are added to or removed from the shelf.
    - **Integration**: The complete system should be capable of running on a cloud platform to allow remote access and testing.
3. **Performance and Reliability**:
    - **Accuracy**: The system should achieve at least 98% accuracy in product recognition.
    - **Response Time**: The virtual cart should update within 500 milliseconds of item interaction.
    - **Robustness**: Include error handling for potential issues such as camera disconnections or network delays.

Deliverables:

- Complete source code for both the machine learning model and the system architecture.
- The system deployed on a cloud server with access details provided.
- Comprehensive documentation that includes:
    - System design and architectural diagrams.

- Detailed setup and testing instructions.
- A description of the machine learning model development process, including data handling, model training, validation, and any challenges encountered.

Evaluation:

- You will be required to present your working prototype at the end of the challenge week. This presentation should include a live demonstration and a detailed discussion of your design choices and implementation strategies.
- A Q&A session will follow, where you will answer in-depth technical questions to assess your understanding and problem-solving approaches.

Timeline:

- The challenge must be completed within one week from the start date.

Please confirm your acceptance of this challenge and the start date so we can prepare the necessary access and resources. We look forward to seeing your innovative solutions and approach to this complex task.