Other Approaches

1. Kafka Approach
   Using Apache Kafka we create a producer/consumer model. We write to a specific topic in the Kafka server using the form API and then create consumers to subscribe to that topic and serve different functionalities. It can be implemented on bare metal as well as in a serverless environment as microservices.
   I selected this approach because it is one of the best methods to deal with this task and it lets me demonstrate my programming skills as well.
   a. Pros:
      i. Scalable, since each consumer uses very minute computing power
      ii. Easy to integrate and implement
      iii. Easy to monitor and manage
      iv. Open-Source
   b. Cons:
      i. Keeping Too Much Data
      ii. Balancing topics is a tedious task

2. Kafka Connect Approach
   This is very similar to the Kafka approach but here we take data once it has reached the primary storage. Pros and cons are similar to the above approach but one more con would be that it will increase reads of the database

3. Serverless Approach
   Using a serverless platform like GCP or AWS to create a publisher/subscriber model.
   Using Google's Pub/Sub and AWS's SNS topics, we can create topics to which the forms will publish. And we can use Cloud Functions and Lambda Functions to subscribe to the said topics and retrieve data to create different functionalities.
   a. Pros:
      i. Extremely scalable
      ii. Easy to integrate and implement
      iii. Easy to monitor and manage
   b. Cons:
      i. Using native cloud services can be expensive

4. API Approach:
   Calling multiple APIs once the form is submitted
   a. Pros
      i. Easy to implement
   b. Cons
      i. Not scalable
      ii. If one API fails there will be a discrepancy