Challenge:

In our bank, we have a continuous flow of data coming from different customer activities. One of the most common ones are credit card transactions. Customers worldwide using their cards generate transaction events.

One of the main challenges we have in the big data and machine learning world is to capture that event in real time, process it, archive it and use it for real-time decision making e.g. using a machine learning model to predict a value.

For this assignment we are looking to replicate a simple event streaming system, a pub/sub step, some simple transformation and then some simple logic.

Using the stack of your choice:

- Build a fake emitter of credit card transactions with random time intervals
- Set-up a pub/sub system to capture those events
- Using PySpark, aggregate all transactions by a given user in the last time interval (e.g. 6 hours) and:
    - Save into a database
    - Open an API that gives the sum of the transactions
    - Displays list of transactions

Deliverables:

1. GitHub link for the core code with:
    a. Emitter
    b. Pub/Sub
    c. API with the sum and list
2. Readme: Description of your solution including
    a. Decision on stack components and why
    b. Description on how the pieces interact
    c. Suggestions on if this were to be moved from a demo to a real system what considerations would you make?

Time allowed:

Around 1 week and to be debrief during Interview (To be scheduled)

Notes:

1. Any web server stack at your own preference

2. Bonus for visualization/dashboard of any kind

3. KiSS Principle, run simple to showcase in your machine or any hosted web notebooks