Immediate Goals:

1. A functioning tool to display how events on Ticketmaster are moving over the course of multiple days
2. Pull Ticketmaster data on inventory and prices for their events on an interval
3. Use the data to guide user action (purchase amount, sale price)
4. Have data visualization (dashboard, excel, etc) for convenient user experience
5. Alerts for certain events (high/low thresholds for inventory, prices) so users can take action

Background

* Current tool was developed over the course of 2-3 months with Ezra as architect and Indian consulting team
* Original estimate was 2-3 weeks for crawling functionality, 1 week for UI
* User count: internally 10-20 users
* Primarily engaged with US, Canada events; a small amount of Mexico

Current State:

* Functionality
  + User is able to select an event and view its historical inventory amounts by day through a line graph
  + Web app
* Code
  + Application uses 2 public TM API’s to get list of events
  + Saves inventory data with timestamps in PostgresDB
    - Pipeline will currently be able to map new event occurrences with any existing series
      * New locations/times/etc
    - Inventory data: dropped into a table
  + Displays data in a graph
  + Crawler hosted in a cloud server
  + Database is hosted in a local server
* Issues:
  + Data variance
  + Data ingestion interval should be once/24hrs but can be erratic (multiple times per day, or 0 times over multiple days)
  + Devs didn't understand big picture and couldn't debug own code
  + Poor code quality, lacking:
    - Organization
    - Conciseness
    - Documentation
    - Structure

Next Steps:

* Background
  + Seeking technically experienced and professionally accountable partner for achieving immediate goals as well as future goals
* Future Goals
  + Crawling additional second-hand marketplaces (Stubhub, Vividseats, etc)
  + Normalize inventory data table
  + Configure/customize data pulling for various events, marketplaces, etc
  + Data-driven predictions on inventory, prices
  + Automated buying, selling