Thank you for your interest in engineering at Outlier! For this UI system challenge, you will have **two days** to submit a written description of your solution.
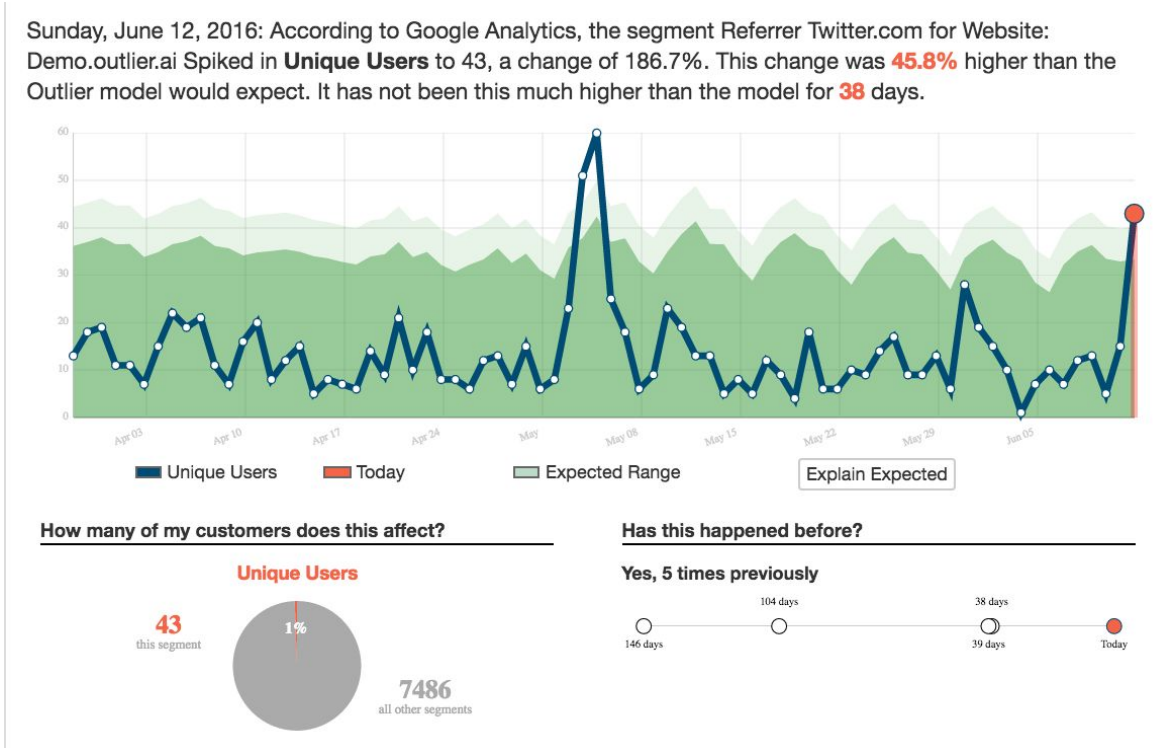
What is Outlier?

Outlier is a new way of thinking about business intelligence. Instead of creating new dashboards or running queries when questions arise, Outlier watches all of your business data for you and lets you know when unexpected things are happening, anywhere in the myriad metrics and dimensions of your data. Outlier plugs into all of your business data where it lives (e.g. Google Analytics, Salesforce, Stripe, SQL databases) and understands your business cycles, trends and patterns for every different segment of your customers. With Outlier, you'll get the headlines of your business each day and never again wonder about what you might be missing.

Problem Statement

Outlier generates stories about your data, which capture insights about your business and provide context to help you make a decision. Your feed is an ordered collection of stories generated for you everyday. A sample Feed may look like:
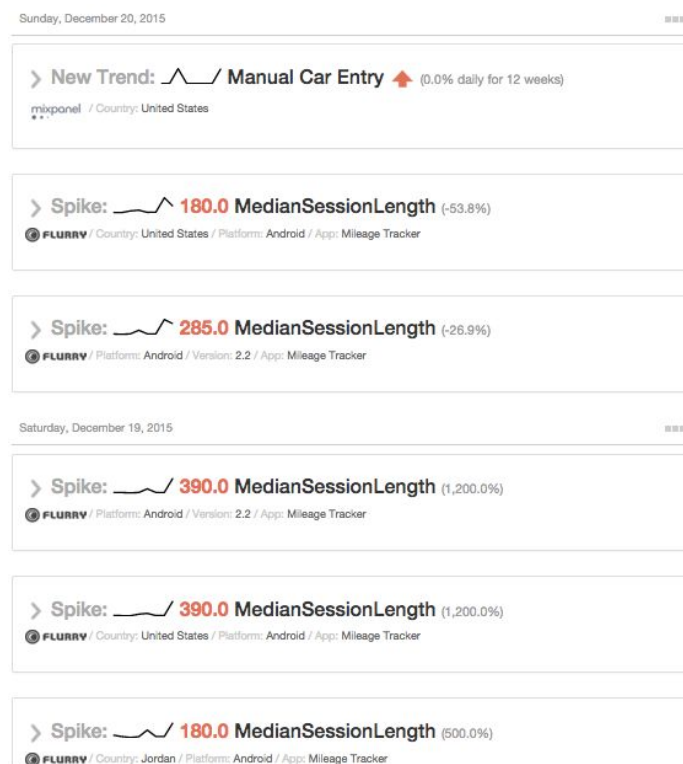
A story summarizes pertinent information on a given metric (e.g. - Sessions) for a segment of your users (e.g. - Referred by Facebook). An example full expanded story might look like:

Sunday, June 12, 2016: According to Google Analytics, the segment Referrer Twitter.com for Website: Demo.outlier.ai Spiked in **Unique Users** to 43, a change of 186.7%. This change was **45.8%** higher than the Outlier model would expect. It has not been this much higher than the model for **38** days.

Assignment

Your goal in this assignment is to **design a front-end architecture that will support the display of the feed and expanded stories**. Your design should provide modularity, reusability, and scale efficiently with the number of stories in the feed. As part of this assignment you should also **design a series of API endpoints** to retrieve the individual elements of the story as presented in the header and expanded story (outlined below). Please note that while your design does not need to adhere strictly to a convention of any particular framework, we do use AngularJS at Outlier.

The discrete elements that will be available in a typical story include (bolded items represent values from the sample story above):

Date / Date Range	**Sunday, June 12, 2016** - The is the day or Date Range (June 12 - 18) a story applies to. A story can apply to day, week, month, quarter or year. For example, Sessions increased 20% over the past quarter.
Story Type	**Spike** - The type of anomaly detected (e.g. - Spike, Drop, New Trend)
Metric	**Unique Users** - The metric or kpi that is the topic of the story. Other metrics include Session, Page View Duration, Custom Events, Page Views, Bounce Rate, etc.

Segment	**Referrer: Twitter.com, Website: Outlier.demo.ai** - The segment of users related to this story (e.g. - Referrer, page path, website, country).
Vendor	**Google Analytics** - The source of the report (e.g. - Google Analytics, Flurry, Mixpanel, Salesforce, Redshift)
Metric Value	**43** - The value of the metric for the given segment.
Percentage Change	**186.7%** - The percentage of the change in a given metric over the previous period. On a daily story this will be the change day over day, in a weekly story this will be the change week over week and so on.
Population impacted	**1%** - The percentage of total users impacted by this story.
Last Occurrence	**38** - Length of time in days since previous occurrences and number of past occurrences since this last story type.
Chart	Visual representation of historical data values for this metric/segment combination. The green area represents ranges the Outlier model expects the data to operate within. Hovering over each dot provide the date and value for that day.

Please feel free to state and make any reasonable assumptions about the interface, and ask relevant clarifying questions. We look forward to reviewing your designs.