Agenda:

* Just status updates from the team

Notes:

* Where the functions/logic is defined (GraphsContainer)
* Ideal data type and format for reading x and y values
* In Graph.js, what graphData[] actually represents
* Get all data into one place a single time (i.e. nested, multidimensional array with sessions containing questions containing etc.)?
  + Something like dot notation to access everything
* Direct access to the database would be useful
  + Possibility to connect to live db via mongo shell or compass?
  + Clone live db to local db
* How to insert data into the Graphs (x’s refer to a particular column/bar, y’s should be either totalled or averaged for a particular x)
* Will probably need to write functions and then call them when a picker option is selected
* Promises - working with asynchronous methods
* GraphsContainer is supposed to populate sessionQuestionsDict - was the intent to have a dictionary for every data type, and is that the best way to group our data to pass into Graphs?
* Redux to handle data?
* Common among us is Ruby on Rails, so much of today’s discussion was compared to workflow and structure in RR terms
* Success working with getter API methods (probably still need to research promises, but was able to access the objects)

Action Items:

* Go through repo documentation regarding database setup/access and confirm that, if it’s possible, it’s okay with Julio
* Research Javascript promises and how to work with them
* Determine data format for working with the values -> getting them into a graph
* Rewrite or fix the nested iterator in GraphsContainer (possibly elsewhere) - sessionQuestionsDict only contains one element (something like {642e245867c86148acef0858: 3}; either we’re handling it wrong or it’s not working as intended
* Look into Redux
* Check-in on Saturday