Meeting Minutes: 10/4/16

What has changed from last meeting:

* Built a separate app, called FaceTracker, which effectively multitracks faces, detects whether each eye is open, and gives a happiness rating based on facial feature
* Renamed our app to RoboApp
* Added a menu button for developers
* RoboApp tells user whether or not commands are recognized
* Continued research in using ultrasonic sensor use with arduino board. Actual implementation details were further clarified.
* UI idea for battery indicator: tired eyes on low battery
* Verified that all voice commands are recognized
* Learned how accelerometer data works

**Immediate goals:**

1. Group -clean out obviously deprecated/unnecessary code as you modify the code
2. Voice Recognition Team
   1. Alex -
      1. Continue making progress on active listening using google API\
         1. <http://stackoverflow.com/questions/16219601/how-to-make-google-voice-run-continually-in-the-background-on-android>
         2. https://developers.google.com/web/updates/2013/01/Voice-Driven-Web-Apps-Introduction-to-the-Web-Speech-API?hl=en
      2. Continue cleaning code
   2. Kurtis -
      1. Continue making progress on active listening
      2. Debug the reason that servos motors don’t work correctly
3. Environmental Mapping
   1. Zach:
      1. Implement battery indicator on the app
      2. Work on android to arduino connection
      3. Work on distance mapping
   2. Seattle
      1. Log accelerometer data
      2. Discover connections between arduino and servos controller by pin number
      3. Work on arduino to servos connection
4. Facial Recognition/color Tracking
   1. Quintin:
      1. Flip facial tracking app to use front camera
      2. Remove old openCV code for facial tracking
      3. Insert new facial tracking code into app
   2. Lukas:
      1. implement color recognition using the android API
      2. Longer term: use legacy color tracking code to inspire how the cat’s head will move to follow colors