Project Proposal/ Ideas:

Team member: Henry, Gini, Panarat, Ekin

**BASE PLAN:** website where user can draw a number/ operator on canvas and machine will predict what user input.

1. *Step 1:* Use the extended MNIST dataset with Kaggle Dataset for math operators (like plus “+”, or minus “-“, or multiplier “ x” or “\*”, etc.) to train model in python.
2. *Step 2*: Save the model (python) and convert it to Javascript TensorFlow model. I’m not sure if it is better to build a flask app to feed the input from client end back to server, let the model predict, then pass result back to client, or just convert the whole thing to javascript and let it handle on the client end the whole time.
3. *Step 3:* Build a webpage with a simple layout like this:

Project Des | About us | Data | Reference

Nav Bar

Masthead / Jumbotron

Output:

What number/ operator is it?

How accurate is this prediction?

Input:

Canvas for user to write on (number/ math operator)

Clear

Submit

1. *Step 4:* Read input canvas pixel RGB value (0 – 255) and make an array (need Jeff help!). Flatten out image to a 784 item array.
2. *Step 5:* Feed the user input pixelated array to model, get the results and display it on the webpage.

**EXTENDED PLAN (if time permits):** A step further

Recognize the number and operator at the same time (input the whole equation instead of just single digits), model will predict what number and operator are input, then do the math. For example, user input 2 + 2 (handwritten), then the machine will interpret the number/ operator, then compute the result.

**Project WorkFlow:**

1. First Meeting: 07-13-2020
2. Python CNN developed and trained data => get the model for predict new input letter from user – Henry -- DONE
3. Flask App: have REST API –Gini
   1. rendering html, css, js -- DONE
   2. 2-way communication (server-client) -- still waiting on Panarat Codes
4. Index.html: the most important part is creating the canvas where user can draw --Ekin
   1. JS: multiple options
      1. How to upload an image to the server (stack overflow) -- Panarat
      2. How to get image in post request (streaming) – Panarat
5. Second Meeting: 07-15-2020
6. JS to get picture data – Henry & Panara (07/17)
7. Post picture data to server - Henry and Panara (07/17- 18)
8. Get prediction and pass it back to html - Henry & Panara (07/18)
9. Index html main page design & format: Ekin & Gini (07/17-18)
   1. Navigation bar
   2. Masthead / Jumbotron
   3. Canvas
   4. Output box / results
   5. Project description (write something about CNN and Dense NN)
   6. About (Who we are)
   7. Data (we use MNIST data)
   8. Refence (What resources that we use, where example codes were downloaded from, books that we read, etc.)

**Worse case::** everything will be ready by 07/20/2020